



KOMIN COMPUTERSYSTEMEN

NEDERLANDS GROOTSTE

SINCLAIR SPECIALIST

7000 TRANSFORMAATBOORD



De 7000 Transformaatscherm is een zeer flexibel scherm dat kan worden gebruikt als monitor voor de Spectrum 486 of de Spectrum 640.

f 225,00

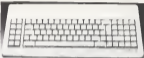
**SAGA 3
ELITE**

Hier begint elke Spectrum 486 of Spectrum 640. Het is de basis van elke Spectrum 486 of Spectrum 640. Het is de basis van elke Spectrum 486 of Spectrum 640.

SPECTRUM HARDWARE

486 Spectrum 486 met 1024K RAM en 10MB harddisk	f 195,00
7200 Spectrum 486 met 1024K RAM en 10MB harddisk	f 295,00
7200 Spectrum 640 met 1024K RAM en 10MB harddisk	f 120,00

54079 Spectrum 486 met 1024K RAM en 10MB harddisk	f 899,00
54079 Spectrum 640 met 1024K RAM en 10MB harddisk	f 449,00



f 205,00

VRAAG ONZE GRATIS CATALOGUS!

7000 TRANSFORMAATBOORD



De 7000 Transformaatscherm is een zeer flexibel scherm dat kan worden gebruikt als monitor voor de Spectrum 486 of de Spectrum 640.



f 225,00

QL HARDWARE

Q120 Omega QL 1 (486/1024K)	f 199,00
Q120 Omega QL 2 (486/1024K)	f 299,00
Q120 Omega QL 3 (486/1024K)	f 399,00
Q120 Omega QL 4 (486/1024K)	f 499,00
Q120 Omega QL 5 (486/1024K)	f 599,00
Q120 Omega QL 6 (486/1024K)	f 699,00
Q120 Omega QL 7 (486/1024K)	f 799,00
Q120 Omega QL 8 (486/1024K)	f 899,00
Q120 Omega QL 9 (486/1024K)	f 999,00
Q120 Omega QL 10 (486/1024K)	f 1099,00
Q120 Omega QL 11 (486/1024K)	f 1199,00
Q120 Omega QL 12 (486/1024K)	f 1299,00
Q120 Omega QL 13 (486/1024K)	f 1399,00
Q120 Omega QL 14 (486/1024K)	f 1499,00
Q120 Omega QL 15 (486/1024K)	f 1599,00
Q120 Omega QL 16 (486/1024K)	f 1699,00
Q120 Omega QL 17 (486/1024K)	f 1799,00
Q120 Omega QL 18 (486/1024K)	f 1899,00
Q120 Omega QL 19 (486/1024K)	f 1999,00
Q120 Omega QL 20 (486/1024K)	f 2099,00
Q120 Omega QL 21 (486/1024K)	f 2199,00
Q120 Omega QL 22 (486/1024K)	f 2299,00
Q120 Omega QL 23 (486/1024K)	f 2399,00
Q120 Omega QL 24 (486/1024K)	f 2499,00
Q120 Omega QL 25 (486/1024K)	f 2599,00
Q120 Omega QL 26 (486/1024K)	f 2699,00
Q120 Omega QL 27 (486/1024K)	f 2799,00
Q120 Omega QL 28 (486/1024K)	f 2899,00
Q120 Omega QL 29 (486/1024K)	f 2999,00
Q120 Omega QL 30 (486/1024K)	f 3099,00
Q120 Omega QL 31 (486/1024K)	f 3199,00
Q120 Omega QL 32 (486/1024K)	f 3299,00
Q120 Omega QL 33 (486/1024K)	f 3399,00
Q120 Omega QL 34 (486/1024K)	f 3499,00
Q120 Omega QL 35 (486/1024K)	f 3599,00
Q120 Omega QL 36 (486/1024K)	f 3699,00
Q120 Omega QL 37 (486/1024K)	f 3799,00
Q120 Omega QL 38 (486/1024K)	f 3899,00
Q120 Omega QL 39 (486/1024K)	f 3999,00
Q120 Omega QL 40 (486/1024K)	f 4099,00
Q120 Omega QL 41 (486/1024K)	f 4199,00
Q120 Omega QL 42 (486/1024K)	f 4299,00
Q120 Omega QL 43 (486/1024K)	f 4399,00
Q120 Omega QL 44 (486/1024K)	f 4499,00
Q120 Omega QL 45 (486/1024K)	f 4599,00
Q120 Omega QL 46 (486/1024K)	f 4699,00
Q120 Omega QL 47 (486/1024K)	f 4799,00
Q120 Omega QL 48 (486/1024K)	f 4899,00
Q120 Omega QL 49 (486/1024K)	f 4999,00
Q120 Omega QL 50 (486/1024K)	f 5099,00
Q120 Omega QL 51 (486/1024K)	f 5199,00
Q120 Omega QL 52 (486/1024K)	f 5299,00
Q120 Omega QL 53 (486/1024K)	f 5399,00
Q120 Omega QL 54 (486/1024K)	f 5499,00
Q120 Omega QL 55 (486/1024K)	f 5599,00
Q120 Omega QL 56 (486/1024K)	f 5699,00
Q120 Omega QL 57 (486/1024K)	f 5799,00
Q120 Omega QL 58 (486/1024K)	f 5899,00
Q120 Omega QL 59 (486/1024K)	f 5999,00
Q120 Omega QL 60 (486/1024K)	f 6099,00
Q120 Omega QL 61 (486/1024K)	f 6199,00
Q120 Omega QL 62 (486/1024K)	f 6299,00
Q120 Omega QL 63 (486/1024K)	f 6399,00
Q120 Omega QL 64 (486/1024K)	f 6499,00
Q120 Omega QL 65 (486/1024K)	f 6599,00
Q120 Omega QL 66 (486/1024K)	f 6699,00
Q120 Omega QL 67 (486/1024K)	f 6799,00
Q120 Omega QL 68 (486/1024K)	f 6899,00
Q120 Omega QL 69 (486/1024K)	f 6999,00
Q120 Omega QL 70 (486/1024K)	f 7099,00
Q120 Omega QL 71 (486/1024K)	f 7199,00
Q120 Omega QL 72 (486/1024K)	f 7299,00
Q120 Omega QL 73 (486/1024K)	f 7399,00
Q120 Omega QL 74 (486/1024K)	f 7499,00
Q120 Omega QL 75 (486/1024K)	f 7599,00
Q120 Omega QL 76 (486/1024K)	f 7699,00
Q120 Omega QL 77 (486/1024K)	f 7799,00
Q120 Omega QL 78 (486/1024K)	f 7899,00
Q120 Omega QL 79 (486/1024K)	f 7999,00
Q120 Omega QL 80 (486/1024K)	f 8099,00
Q120 Omega QL 81 (486/1024K)	f 8199,00
Q120 Omega QL 82 (486/1024K)	f 8299,00
Q120 Omega QL 83 (486/1024K)	f 8399,00
Q120 Omega QL 84 (486/1024K)	f 8499,00
Q120 Omega QL 85 (486/1024K)	f 8599,00
Q120 Omega QL 86 (486/1024K)	f 8699,00
Q120 Omega QL 87 (486/1024K)	f 8799,00
Q120 Omega QL 88 (486/1024K)	f 8899,00
Q120 Omega QL 89 (486/1024K)	f 8999,00
Q120 Omega QL 90 (486/1024K)	f 9099,00
Q120 Omega QL 91 (486/1024K)	f 9199,00
Q120 Omega QL 92 (486/1024K)	f 9299,00
Q120 Omega QL 93 (486/1024K)	f 9399,00
Q120 Omega QL 94 (486/1024K)	f 9499,00
Q120 Omega QL 95 (486/1024K)	f 9599,00
Q120 Omega QL 96 (486/1024K)	f 9699,00
Q120 Omega QL 97 (486/1024K)	f 9799,00
Q120 Omega QL 98 (486/1024K)	f 9899,00
Q120 Omega QL 99 (486/1024K)	f 9999,00
Q120 Omega QL 100 (486/1024K)	f 10099,00

495

**QL
OMEGA**

QL Omega QL 1 (486/1024K)

QL Omega QL 1 (486/1024K)	f 199,00
QL Omega QL 2 (486/1024K)	f 299,00
QL Omega QL 3 (486/1024K)	f 399,00
QL Omega QL 4 (486/1024K)	f 499,00

De QL Omega QL 1 (486/1024K) is een zeer flexibel scherm dat kan worden gebruikt als monitor voor de Spectrum 486 of de Spectrum 640. Het is de basis van elke Spectrum 486 of Spectrum 640.

BINNENKOMT LIEVERBAAR: CST THOR QL QL OM
PAYABLE EN IN LOK: OPLOSSER VAN DE SA
CLAMP QL INFORMATIE BEZICHTBAAR

CST THOR

WIJZIGINGEN VOORBEHOUDEN

Onze showrooms zijn geopend
maandag van 9 tot 5 uur.

INFORMATIE EN BESTELLEN
KAN OOK TELEFONISCH
TE WERKEN
25 000 0000

Postbus 1600
5000 CA Oirschoten
De Oirschoten 15a
5000 CA Oirschoten
Telefoon 040 4100000
Telefax 040 4100000
Bank: Rekening 15 00 00 701
A/RN 50 00 70 011
Oms: 00 43 001

VERZENDINGEN ONDER REMBOURS OF BIJ VOORSCHUUTALING

sinclair

NUMMER 2

PRIS F 6,50 / 130 BF

gids

ONAFHANKELIJK BLAD VOOR SINCLAIR GEBRUIKERS

SERIES:
SYSTEEMVARIABLEN
TASWORD

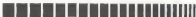


TEST: PSION ORGANISER

LISTINGS VOOR SPECTRUM

SOFTWAREBESPREKINGEN

COMPACTER VOOR QL

**new****COMPLIMENTEN
VAN SAGA**

Saga was tot nog toe bekend door zijn games van vervaagende toetsenborden voor de Spectrum. Het lijkt er op, dat ze van dit laagje afstijgen, en op een bredere markt stikken.

Saga Systems brengt een compleet pakket op de markt, voor de Spectrum 48K. Dat pakket heet nu "Compliment". Het bevat alles om van de Spectrum een volledig, kleine professioneel systeem te maken.

Als hardware :

- 1- een printer die 120 tekens per seconde afdrukt, en Letter Quality (geen naar meer); een buffer van 2 K; parallelle interface (Dma); eigen interne/externe karaktercode; printkop met 9 rastering; bidirectioneel printen.
- 2- een Saga 2- toetsenbord met quasi rasterlijk blok toetsenbord voor dekl ingeven.
- 3- een Opus diskdrive, nu al ongeveer de standaard diskdrive voor Spectrum. Compleet compatibel met microdrive-MSDC, dus alle MSDC programma's kunnen zonder meer gebruikt worden.
- 4- parallelle interface voor de printer
- 5- voeding ingebouwd in 1 rastering voor het hele systeem (behalve printer)
- 6- joystick poort

Als software :

- 1- de woordprocessor van Saga zelf, The Last Word, zeer goed vergelijkbaar met Word 3.
- 2- Brats Express, de utility om cassette-software op Opus disk over te zetten.
- 3- een database die massaal inge gegevens heeft te worden; Masterfile 4- een spreadsheet die ook al lang heeft bestaan wat hij kan; Calcula
- 5- spreadsheet, om mooie schermen op papier te zetten
- 6- programmeerhulp om toetsen te herdefiniëren

Het lijkt ons dat Saga hiermee een goed doet naar de markt van waarvoor de, wat de PEEKS woordprocessor. Er zijn ons nog geen prijzen voor België of Nederland bekend. In Engeland kost hij £248 inclusief. Dat is alvast 100 pond goedkoper dan de Amstrad, die dan nog massaal voor iets anders dan tekstverwerking is te gebruiken. (P.P.)

**Leerling-tenaar
The Discipline**

Wellicht wat te laat voor de 48K, maar net op tijd voor de 128K en de +2. Brengt Rockford Products een veelzijdige interface op de markt : de Discipline. Als mogelijke voorzieningen die je kunt kiezen, zijn ingebouwd.

De Discipline beschikt over een disk-interface waar je alle kanten aan uit kunt : single of double density, eenzijdige of dubbelzijdige diskettes, 40 of 80 tracks, 5 inch, 3,5 inch, 5.25 inch, alles is mogelijk. De bediening gaat via eenvoudige MSDC command's. De laadgesnelheid ligt rond 1/3-2 seconden voor een "volle" Spectrum 48K.

Handig is de automatische back-up knop, waarmee de gegevensinhoud van de Spectrum op schijf wordt gezet. Handig om langdurige of erg moeilijke spellen later verder te spelen vanaf het punt waarop ze werden afgebroken.

De printer-interface is Centronics parallel, dus standaard. Hij sluitert naar de gewone command's LPRINT, LPRINT en ook GOF's. Hij versorgt eenware grafische tekens.

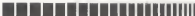
Er zitten ook twee joystick-aansluitingen op, die met Sinclair en met Kespcon joystickke kunnen werken.

De interface kan ook gebruikt worden als netwerk-station. Vervolgens Spectrum kan ook gewoon (met) kunnen op die manier "mogelijk" gebruik maken van double printer of van double disk drive. Dit netwerk zou ook computabel zijn met de interface 1) wij vermelden dat daarmee bedoeld wordt, dat beide interfaces met elkaar kunnen communiceren, en niet dat ze mogelijk aansluitbaar zijn.

Het keetje ziet er uit zoals de interface is, en sluit ook op double en/of achterop de Spectrum aan. Het is niet goedkoop, vinden wij, maar het kan een goed alternatief zijn voor de Opus, indien je een goedkope disk drive op de leg kunt vinden. (P.P.)

Rockford Products, The Discipline
Prijs : £94.75 inclusief

Saga Systems Ltd, Compliment



new

**SANDY B-MT440
3 THOR**

Net als eerder al melding hebben kunnen maken van de CDT THOR heeft nu SANDY de zijn opvolger van de klasse. SANDY geeft bij de klasse van B-MT440 uit een andere principes dan CDT.

CDT maakt van de GL een machine gericht op zakelijk gebruik. SANDY richt zich meer op de GL-gebruiker. Dit verschil komt ook duidelijk tot uitdrukking in de prijs. De B-MT440 is in Engeland met 3 drives zo'n 150 pond meer dan de THOR met twee drives. Bovendien blijft bij de SANDY een microdrive gehandhaafd.

Bij de THOR is er de mogelijkheid om je eigen GL te laten bouwen. Bij SANDY koop je een kant-en-klaar machine of een Do-it-Yourself pakket.

Als je al een CDDSD van SANDY hebt is de keuze niet zo moeilijk. Je kunt dan volstaan met een DO-pakket van Gb. 45.000.

De CDT THOR is inmiddels bij beide exporteurs geïmporteerd. Op de eigenlappen HCG-roepen kon worden met Eurodrive met trots al het eerste exemplaar afgeleverd. De grootste verandering in de prijs:

De THOR met 3 drives 41.000 met twee drives 34.000

Een stuk lager dan waarschijnlijk gedacht werd. Een fijne verandering. De THOR doet er uitzien als een klassiek stuk. Een toestel dat je vingers bij af te likken.

**GL - DESK TOP
PUBLISHING**

Desk top MP Software is het voortaan ook mogelijk om met de GL een desk top publishing te doen "letters" op het eigen kantoor.

Het programma is volledig aan-gestuurd. Het laat toe de te knippen en te plakken, te schrillen, stukken apart te bekijken, de lettergrootte te wijzigen, en zelfs eenvoudige tekeningen te maken.

Het programma laat toe de zelf karakters in een 8x8 raster te definiëren en te tekenen met de wijze hand te maken. Het beschikt over een tweede karakter-set en print op 80 regels van 132 kolommen. Het kan 48 bladzijden op een Epson-compatible printer afrukken.

Het idee van het programma is, dat het de helft van een blad in het gekozen formaat, en de andere helft op microdrive. Het is ook mogelijk de een ROM-geheugen te gebruiken, als je sneller wilt werken. Je kunt op alle soorten kleine letters een van de gekende spelling-mogelijkheden.

(P.P.)

Inlichtingen:
Nederland:

KORIN BV, de Graafstr. 15A
Eindhoven, 040-484400

Belgie:

Micro-Conversion
St-Pieterstraat 18-19
Antwerpen 200-3349400

**PROPERE IONER
TOETSENBOORD****GL - HIRNCEL
NINE-NODES**

Hirnzel System brengt een model uit voor de GL. Of hoe een klein ding, groot kan zijn: 70x40x24 en als de reden van het design. De model vertikt tegen 1200/70 baud of 1200/1200 half-duplex. Op een meegleverde microdrive cartridge staat de software op de GL als standaard ter beschikking, je als een 80 koloms scrollende terminal. Verder is communicatie mogelijk met Prentel en Microport, zowel als met diverse niet professionele bulletin boards. De software komt van goede. Voor de prijs hoef je het niet te laten: 49 pond.

(P.P.)

Voortaan is het ook mogelijk om met de GL wat een letterverwerking te doen. Schien Hyncoons brengt het eerste vervangende toetsenbord voor de GL op de markt. De dagelijkse schrijfmachine toetsen maken het gemakkelijk om aan te werken. Het nieuwe toetsenbord is helemaal gebaseerd op de gekende GL. Het is daarom compleet compatibel met alle bestaande software en rand-apparatuur.

Op vijf minuten, volgens Schien, wordt het toetsenbord op de GL geïnstalleerd. De schikking van de toetsen is praktisch helemaal identiek als die op het originele toetsenbord, zodat het erg snel werkt.

Voor 54,90 pond ook je nu van je GL een schrijfmachine. Het voor Karsten?

(P.P.)

RAMPRINT
RAMPRINT

RAM Electronics Ltd. brengt een Centronics printer-interface op de markt. Toen ik het hoorde, dacht ik: "Het is onderhand de bestende interface voor de Spectrum."

Toch het ding meer eens geprobeerd. Het werkt heel handig. Je print het ding op de expansiebuis van de Spectrum (40n, de 4, de 128 k en de 40 n de kabel in de bus van je operationele printer, en klaar is Kees. Bij mij doet er dan nog wel een Quas drive-er tussen de Spectrum en de interface, maar daar trof de RAM PRINT zich duidelijk niets van aan.

Het kan wel overbodig lijken, met die Quas er tussen, omdat die uiteindelijk zelf al een parallelle interface heeft. Maar de RAM PRINT is niet dat enkel een printer-interface. Je kunt er ook erg gemakkelijk een keypad het je tekst er op papier niet uitkomen.

Ook het commando LPRINT "hi hi" krijg je een lijn voor op je scherm, dat bestaat uit de vijf meest nuttige tekens in verband met de printer te bepalen.

1. Je bepaalt of er een "line feed" moet volgen na een bepaalde regel of niet; dat is afhankelijk van je printer. Sommige printers doen dat automatisch, andere niet.

2. Je bepaalt of de interface de niet-ASCII codes geeft als code doorgeeft, dan wel als ASCII-code (ASCII-gecodeerd, Parallax v.w.). Het interface 1 of Quas kan dat door de streep voorafgegeven je print, respectievelijk aan het "b" of het "t" knopje te koppelen.

3. Het is de schere-inhoud op papier hebben (COPPI) dan kan dit ook op twee manieren. De meeste writer werkt enkel met tekst, met ASCII-tekens dus. De "trager" writer = overigens ook nog behoorlijk snel - stuurt alles door in tekensetjes, 106's, naar de printer.

4. Het is de tekst voor lezen ook, de tekst groot afdrucken, dan kan je dit ook bepalen via het kleine menu. Het commando geeft dan wel voor de hele tekst, en niet enkel voor een stukje of zo. Dat kan erg handig zijn voor gebruiksgroepen. Bijvoorbeeld de zelf afdrucken op dit moment te drucken.

5. Het laatste wat je kunt kiezen, is de breedte van de bladzijde. Dat is het aantal karakters dat per regel wordt afgedrukt. Na dit aantal wordt de printer op de lijnruimte gebracht, klaar om de volgende regel te beginnen.

Tot daar de printer-interface. Handig, zonder meer.

Met de RAM PRINT aantrekkelijker maakt dan andere Centronics interfaces, is de ingesloten tekstverwerker. Dat is natuurlijk helemaal afhankelijk. Maar dat de tekstverwerker in RAM zit, heb je operationele toegang, zonder teveel de leedproeven of met dan ook. Gewoon het commando LPRINT "hi word" en je kunt aan de gang met je brief.

Het scherm van de tekstverwerker is opgedeeld in twee zones: één voor de tekst zelf, en één voor het intypen van commando's, waar ook de boodschappen van het programma zelf worden in afgedrukt. Tussen beide zones staat een regel, waarin steeds wat informatie wordt gegeven over de toestand van het programma: of de hoofdletters al dan niet "aan" staan, of je tekst kunt voortzetten of overbrengen, hoe lang een bladzijde is, en hoeveel karakters per regel het "tekstreef" is, waarin je de tekst kunt bekijken.

Het kleine menu heeft behoorlijk wat mogelijkheden. Je kunt tekensetjes kiezen, woorden openen in de tekst, woorden vervangen door andere woorden of zinnen, blokken tekst wissen en tekensetjes verslaten. Ook de manier van afdrucken kan naar aan worden geregeld: eenzijdig, dubbelzijdig, extra fore (nog te maken), de rechtermarge wijzigen, met een bepaalde marge, gemiddelde pagina's, tekst onderbrengt, dat zijn standaard mogelijkheden. Daarboven heb je nog de mogelijkheid om alle andere codes die deze expansiebuis printer nog kan kennen, in de tekst op te nemen. Dat die mogelijkheid werkt voortreffelijk.



test

PSION ORGANISER II



Met heeft de PSION ORGANISER nu het Sinclair te maken met U terecht definitief. Eigenlijk MOET u toch een keer ALLES in 's kort en jk is U verteld en waarom wel.

In de eerste plaats omdat PSION de beste hulpmiddelen van Sinclair waren en daarom later bekend zijn bij Sinclair gebruikers dan bij gebruikers van andere computers. En in de tweede plaats omdat de Organiser eigenlijk een MSX2 van communiceren met de GL. De kort U zien uit Archive kijken in de Organiser. ARCHIVE is bovendien ook een product van Psion. Deze twee overwegingen waren voor mij voldoende en ook wil ik graag bij de Organiser II.

Wat is de ORGANISER II voor 's apparaat?

De ORGANISER is een rekencomputer, een rekenrechner en een elektronische agenda met alarm in een.

Al deze functies zijn ondergebracht in een 32K ROM (model 8F5). Daarnaast beschikt de organiser over 14K RAM (8F1).

Wij hadden voor deze bespreking de beschrijving over een model XP. Dit werd later ter beschikking gesteld door de importeur voor België: Micro-Connect van Antwerpen.

Maakt het model XP is er nog een model CP. Het verschil zit hem in de geheugen capaciteit.

De XP is het topmodel, dat twee "data-pak" van maximaal 128K kan gebruiken.

De CP is het basismodel, dat twee datapak van maximaal 64K kan gebruiken.

De ORGANISER is een zeer krachtig apparaat dat maximaal 304K RAM geheugen kan gebruiken. Het is dus het krachtigste rekencomputer, die ik tot op heden in m'n vingers heb gehad. Een PC is tekortkomek."

Hoe ziet de ORGANISER eruit ?

De afmetingen zijn opvallend klein voor zo'n krachtig apparaat. Looft 142x76x29 mm. Weegt slechts 250 gram. Het apparaat wordt gevoed door een 9V alkaline batterij en dat is voldoende voor 3-4 maanden werken.

De centrale processor is er een uit de 8080 familie met 130K2 CPU en 8 bits. Het scherm is een dot-matrix LED en bestaat uit 2 regels van 16 karakters. Dat met moet zeggen dat je geen langere regels kunt gebruiken. Regel twee van de display is namelijk zo geschikt dat deze automatisch gaat rollen als de tekst langer is dan 16 karakters. Het rollen kan gecontroleerd worden met de cursor-toetsen.

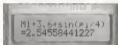
Het toetsenbord bestaat uit 36 toetsen: 10 numerieke toetsen en 16 reken- en functietoetsen. Bovendien is het geheel afdekkt in twee gedeeltes als kleine letters onder deze 36 toetsen aanwezig is.

Een ik ga niet aan de korrelrijke geeft U de mogelijkheid de printer, knoppen of andere aan te sluiten.

Aan de achterzijde vindt U twee kleine aansluitingen. Het eerste is een "dataport" met aansluiting voor het aansluiten van al de gegevens. Het tweede is dat een dataport in een van de EPROM is van 16, 32, 64 of 128K. Dus als U het apparaat wilt blijven de gegevens bewaren.

Gaat de dataport zijn er ook verblijven te koop met speciale programma's, zoals ARCHIVE voor allerlei administratieve besparingen en FINANCE een pakket waarmee allerlei financiële zaken kunnen worden bijgehouden.

Interessant kunt U zelf een programma schrijven en dit in een datapak opslaan, daarna heeft U de eigen programma altijd direct ter beschikking.



ANDEREGENTIEN

Voor de Eigenaar is inmiddels ook de enige randapparatuur voorhanden. De gewone een opvoeding met de de dit aantal bekende prijzen. De prijzen in Nederlandse guldens zijn onder prijzen.

100 datapak	1295 F/ ca. 43 70
100 datapak	1295 F/ ca. 43 110
200 datapak	2495 F/ ca. 43 220
400 datapak	2495 F/ ca. 43 440
1200 datapak	12995 F/ ca. 43 710

FINANCE Program Pack	2495 F/ ca. 43 145
MATHS Program Pack	2495 F/ ca. 43 145

PC232 verbinding	4795 F/ ca. 43 260
Local Networking	1295 F/ ca. 43 70



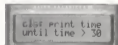
Voor professionele toepassingen zijn er ook nog een IBM CGSE linker en een linker voor Magnetische Kaarten leverbaar. Voor inschrijvingen wordt een rich tot de ingetast.

PROGRAMMEREN

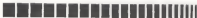
Voor het zelf programmeren heeft U de beschikking over een BASIC-versie, dat POIN de DPL heeft gekocht. DPL staat voor Operating System and Programming Language. De programmeert taal het gestructureerd programmeren toe, hoevel procedures niet tot de DPL komen.

De andere bij het programmeren voor het toetsenbord van de Eigenaar. Dit staat het vlot programmeren duidelijk in de weg. Als je een computer geest bent kun je daar echt een afzien. Het kijkt voor al wel gelden dat alles went, maar toch een uitgang.

Voor het werken met grotere systemen heeft PC232 voor de IBM PC, AT en 486 een emulator geschreven, zodat de professional nu gebruik kan goede ondersteuning krijgt. Welke voor 486 bezitters voor de 486 is er geen emulator anders had U ook op een eenvoudige wijze kunnen programmeren met de Eigenaar. Het blijft voor de gebruiker de mogelijkheid om files uit WORDS van de Eigenaar toe te vertrouwen.



De command's en functies die U binnen DPL ter beschikking hebt zijn voldoende om een gestructureerd programma te schrijven. U heeft niet minder dan 110 keywords ter beschikking.



CONCLUSIE

In het kort hebben we gepraat over de U van nieuw te geven van deze werkelijk krachtige rekencomputer. Later voor GLU beschrijven het een redelijke achtergrond van de Organiser met betrekken.

Een adresomstand bij de hand waar U ook een overzicht van de mogelijkheden van de gegevens bestanden van de GLU in te zien. Verder zijn er voor mogelijk gebruik vele randapparaten leverbaar.

We willen de Organiser niet direct als een Yuppier-artikel betrekken, zoals we al gezien hebben in een ander tijdschrift, maar meer als een logisch verlengstuk van de eigen computer. Later op zal zijn gebruik moet data Organiser omgebende mogelijkheden. Het kan komen van de Organiser karakteristiek als een PU in te formaat. En dat laatste kunt U letterlijk lezen.

Hoewel het apparaat niet direct uitnodigt tot zelf programmeren is het een heel makkelijk wonder. Vooral in de laatste twee biedt de ORGANISER omgebende mogelijkheden. Denk daarbij aan het registreren van artikelen enz. Een Code lezer. Noteren van instellingen door verkopers, die deze later via een ander apparaat naar het U. Het noteren van afspraken met alarmmogelijkheid etc.

De prijzen tussen U met ingevallen als U de vergelijkt met normale computers. Door de geavanceerde technieken die zijn toegepast (ENR en de EPROM chips) liggen de prijzen hoger dan U het nu toe gezien had.

DPL KEYWORDS

Keywords voor programmeren:

IF, ELSEIF, ELSE, ENDF
DO, UNTIL
WHILE, ENWH
BREAK, CONTINUE
GOTO

Variabelen:

EXTENT lokale, globale, enkele en arrays.
TYPE Real, integer, string, ijzide

Commands:

SEE IN SOURCE CODES

AT I,j positie op twee schermrijen
BEEP I,j Beep met duur I, freq. j
CLS Wist het scherm
CURSOR I Cursor naar I
BEST str Het een string

ESCAPE ON/OFF Breakmogelijkheid aan/uit
INPUT x/a/a Input van een integer/geheel of tekststring.
SETATE i Het de toestandsetatie
DATA LABEL Ge naar label bij fout
DATA OFF Het foutoplossing uit
Scheel de Organiser uit
MACTIOL n Machtiged in 1 tot een
FOLIE j op adres i Folia j op adres i
PRINT str/a/a Print tekst of getallen
LPRINT str/a/a Print naar de printer
PAUSE i Geef pauze van i
RAISE i De Random generator
RAND n Random command
RER Het 1/a/a/a terug naar de
call procedure
STOP Na uit de DPL
TRAP Onderbreng fouten in het
volgende command

File behandelings:

APPEND Kopieer een record
COPY Kopieer een file
CREATE Creeer een file
DELETE Verwijder een file
DELTE Het een record
ERASE Het een record
FIRST Het eerste record
LAST Het laatste record
NEXT Het volgende record
BACK Het vorig record
OPEN/CLOSE Open of sluit een file,er
kunnen aan. A file tegen
zijn open zijn.
POSITION Selecteert een record op
nummer
RENAME Herbenoem een file
UPDATE Werk een record bij
USE Verander van file

Functions:

FIND(a,b) Vind record met a,b
DATE Geef huidige string
DAY Geef vrs j gevegen
COUNT Geef aantal record/file
POS Geef het huidige recorden
SPACE Geef vrije ruimte in data
pak
DIR(a,b) Geef file naam
FIND(a,b,i,j) Geef a als een getal in
het door i en j bepaalde
formaat
GEN(a,b) Geef a in het heel edge-
lijke formaat
NUM(a,b) Maak van a een integer
SCN(a,b,i,j) Geef a in wetenschappelijk
formaat
ERR Geef de laatste foutcode
ERR(a,b) Geef de foutcode van i
ERR(a,b,a) Toont een record en schere
VER(a,b) Voorzet in Remington
VER(a,b,a) Toont een scrollend string

Vander aanwezige functions

SECOND, MINUTE, HOUR, DAY, MONTH, YEAR
ABS(a), ATAN(a), COS(a), SIN(a), EXP(a),
FLTD(a), LN(a), LOG(a), INT(a), MOD(a),
LN(a), LOG(a), PI, RAD(a), SIN(a), COS(a),
TAN(a).
SET, GET, KEY, KEY

lees verder blz. 45



AMX mouse

Het verhaal over het feit dat de Spectra een staartje kreeg. Daarmee opgelost. MUSEN zijn in deze dagen.

Inmiddels slinger men tijds in de handel in de AMX-MOUSE van de Firma Advanced Memory Systems Ltd. uit Engeland.

De AMX-MOUSE is geschikt voor de Spectra 486 en 143. Het complete pakket rond deze muis bestaat uit: de muis zelf met een crinter-face. Op deze crinter-face kunt u de muis aansluiten, naar behoeven een grafisch printer. Een zeer eenvoudige handleiding, zodat je dan dit soort vermenigvuldigen van het de muis gemakkelijk beheren kunt maken. Daarnaast is er de besturingssoftware en een draaiel hulpprogramma's, t.w. AMX AMI Colours Pakket en ICOM Bebilder. Een 32 pagina's tellende handleiding compleet het geheel.

De MOUSE is een zeer handig apparaat, iets goed gebruikt en wanneer de programma's die met de muis werken, goed geschreven zijn. De mogelijkheden zijn dan legio.

De Firma DATACIP te Breda, van wie wij de AMX-MOUSE voor deze bespreking ter beschikking kregen, heeft niet veel nog meer ten overvloede hoe handig die muis te gebruiken is. Hetzelfde heeft aandacht een aantal programma's rond de muis op de markt gebracht, als D. BASS, AMI, TOPDOWN AMI, CLIPPAKET AMI en de DESKTOP MOUSE voor de CPU's.

Een goed programma met een goed behuizinge muis kan binnen een programma veel bijdragen leveren. Een goed voorbeeld hiervan vormt het programma DESKTOP MOUSE voor de CPU's. Hetzelfde, met een simpele druk op de knop, alle files van de schijf in beeld, die dan weer op hun plaats met een simpele druk op de knop in te laden zijn. Vergelijk de twee drukken op de knop met de vele handleidingen die je anders zou moeten doen om hetzelfde resultaat te krijgen.

De muis ligt goed in de hand en reageert goed. Het is het aan te raden niet te hard op de muis te knopen met de hand, want dan gebeurt het nogal eens dat je niet reageert. Daarom luidt de muis de gele knoppen en er zijn geen problemen.

Kijk voor de muis een mooi glad en vooraf schone oppervlak. Want letsel, en e.d. kunnen zich aan de knoppen hechten en dat kan dan weer storingen veroorzaken. De knop is vrij simpel aan de onderkant van de muis te verwijderen voor een schoonheidsbeurt.



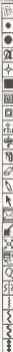
WAT DOET EEN MUIS ?

De handelingen onder ons muis naar schiedelijk nu in gedachten. Het zeggen. Met de muis kunt u met grote uitgesloten cursor (een ICOM) beelden. Met de ICOM, die allerlei vormen kan aannemen, kunt u over het scherm bewegen. Het bewegen doet u met de muis op de tabel naast de computer. De bewegingen die u met de muis maakt worden op het scherm gevolgd door de zich verplaatsende cursor. De bewegingen volgen simpel en glad. Daarbij is de snelheid waarmee de cursor zich verplaatst variabel.

Alleen een bewegende "muis" op het scherm is natuurlijk niet bijzonder. Maar als u daarbij opt. ICOM's (paar) gebruikt kunt gebruiken, dan wordt het systeem al bruikbaar. Meten en daarvoor nog gebruik van de muis. "FULL-ICOM MOUSE" dat is het systeem compleet.

De ICOM's moeten duidelijk verhandelen zijn van hetgeen u op het scherm of op het scherm. In de aangeleverde programma's kunt u daar welke knoppen van een. Voor een ieder van de muis zelf ICOM's aan te wijzen een goede raad is die een goed te gebruiken. Een goed programma ligt aan de hand van de muis. Het is ook buiten op straat of in openbare gebieden een goed raad, je wilt verstandig staan hoevel programma's er al in gebruik zijn. Laatst dus dat deze techniek nu ook bij computers wordt toegepast.

De "FULL-ICOM MOUSE" zijn natuurlijk helemaal het einde. Je hoeft een muis op als je nodig hebt en het is daar onder het scherm gaat niet meer. Laat je de laatste knop waar het is de computer. Het scherm weer afzet en kun je weer verder met je werk. Heten maken uit de muis is ook een fluitje van een cert. Je hoeft niet de muis een lange de diverse opties en het de muis



one-liner competitie

PROGRAMMA'S VAN LEZERS

In deze rubriek zullen wij in elk nummer een aantal programma's vinden, die door actieve lezers werden ingezonden. Voor alle gepubliceerde programma's zullen wij u natuurlijk bedanken, want benutten zij u nog steeds "Voor wat, loont wat". Uiteraard moeten de programma's voldoen aan de eisen, die in de colofon te vinden zijn.

Afhankelijk van de lengte van het programma zullen wij de gepubliceerde inzendingen betalen met een bedrag van resp. fl 15,00, fl 25,00 of fl 40,00. Wij doen ons best om zo veel mogelijk programma's te publiceren. Uiteraard zal de kwaliteit bij het publiceren van het programma natuurlijk een belangrijke rol spelen.

INZENDEN VAN PROGRAMMA'S

Wij vragen wij de lezers, die programma's inzenden voor eventuele publicatie, rekening te houden met het feit dat de redactie geen tijd over heeft om de programma's stuk voor stuk in te kijken. Voor kleine programma's is dat nog wel te doen, maar voor programma's van enkele kilobytes wordt dat echt een keसर.

Daarom een vriendelijk verzoek om de programma's in te zenden op cassette / microdriveschijf of diskette (DPL01). Daarnaast zou het fijn zijn om te weten wat het programma hoort te doen en hoe ze er mee moet werken, zodat we de uitleg erbij kunnen plaatsen. Deze uitleg mag u op papier aanleveren, op cassette, schijf of diskette als TADMOD-file of Quill-file.

Het merkt voor de redactie het best. Nog beter is het natuurlijk als de uitleg in het programma zelf is opgenomen, voor gebruikersprogramma's is dat eigenlijk het beste.

We hopen dat we vele SINCLAIR gebruikers in deze rubriek mogen inspireren. Niet alleen SPECTRUM gebruikers maar ook de gebruikers van de "good old" Z80 zijn van harte welkom en uiteraard de vele QL-gebruikers.

Vooral deze laatste groep heeft het laatste jaar weinig kans gehad om te laten zien wat ze allemaal kunnen op het voortreffelijke QL. Het de hoop van de meeste SINCLAIR GIDS hopen we uiteraard dat deze programma's heeft geschreven voor zijn/haar SINCLAIR computer meer een mede te bieden, om te laten zien, waarom hij/zij in staat is.

ONE-LINER COMPETITIE

Speciaal voor de SPECTRUM gebruikers, is deze ONE-LINER COMPETITIE ingericht.

Zoals de doorgevoerde spectruagebruiker weet is de spectrua een van de meeste computers, die programmeers van "one-partje langte" accepteren.

Het moet voor de spectruagebruiker een "oud kunstje" zijn om een programma te schrijven dat uit slechts 1 programmeergat bestaat.

Maar wij van de redactie wil deze willen weten waarom deze lezers in staat zijn, zijn we met deze ONE-LINER COMPETITIE gestart.

Voor de beste one-liner van elk nummer stellen wij een bedrag van fl 25,00 beschikbaar en voor de dienaar die als tweede eindigt een bedrag van fl 15,00.

REGLEMENT

Uiteraard zijn er voor deze competitie een aantal regels:

1. Alle lezers van de SINCLAIR GIDS mogen meedoen.
2. De ONE-LINER moet in SINCLAIR BASIC geschreven zijn, dus geen extended BASIC programma's.
3. Om te aanmelding te doen voor publicatie moet het programma binnen zijn voor de einddatum van dat nummer (zie tabel). Alles wat daarna binnentkomt schuift door naar het volgende nummer.
4. Het programmaatje moet op cassette worden ingelezen, eventueel met uitleg hoe het programma te kunnen werken.
5. Alle binnengekomen programma's dingen mee naar een der juryleden zijn ook nog bronsprijzen in de vorm van speelprogramma's en boeken.
6. Als de ruiter het toekent zullen wij de ONE-LINER allemaal publiceren in de SINCLAIR GIDS.
7. Alle ingezonden programma's dienen eigen werk te zijn. De deelnemers vrijwaren de SINCLAIR GIDS voor eventuele aanspraken van derden op het copyright van de ingezonden programma's.

Maar regels zijn er niet, dus wat het is de GIDS aan het werk te gaan en mee te doen.

We hopen met deze competitie het talent, het origineelheid schenken is, te krijgen tot toepassing.

Gedat door een medewerker het eerste nummer niet direct kan verschijnen in de volgende lag, hebben we voor dit nummer nog geen one-liners ontvangen. Maar het volgende nummer zullen er mogelijk al een aantal deelnemers zijn binnengekomen.

Veel succes allemaal!

QL

FILECOMPACTER

(Dennis J.)

In dit artikel wordt een programma beschreven dat in staat is om files kleiner te maken. Hiervoor wordt gebruik gemaakt van een variabele lengte code om de letters in de file te representeren. Karakteren die niet zo vaak in de file voorkomen krijgen een langere code dan vaak voorkomende karakters. Uiteraard is ook voorzien in een programma dat de originele file weer terug kan brengen. Aanstondig voor deze programma's verscheen een artikel in Byte mei 1988.

Weg je het nu van dit laten kregen van files? In de eerste plaats schijnt het natuurlijk gekkegerenue, zodat er meer op een contrade of flappe valt te propen, en in de tweede plaats is wordt noemde de transmissie tijd. Daarom als de file over een langere verbinding verstuurd moet worden, bijvoorbeeld over een 300 baud telefoonlijn.

Een file compacter maakt gebruik van Huffman codes. Een Huffman code is in 1952 een manier om uitgaande van een alfabet waarvan voor elke letter de frequentie van voorkomen bekend is een code te maken waarbij de lengtes van de binaire codes niet voor alle letters gelijk zijn en waar juist zodanig dat de code 'optimaal' is. Die literatuurverwijzing 013 voor het bewijs dat die code inderdaad optimaal is. Dit betekent rust, dat we hiermee de best mogelijke compactie methode hebben. Uiteraard slaan we nog vrij in de keuze van het alfabet (letters, lettergroepen, woorden, zinnen, zinnen etc.), en ook zijn er natuurlijk methodes te bedenken die andere werken dan frequentie afhankelijk code systemen. (Charaktere karakters, gebruik maken van het feit dat een er voor iedere letter een andere waarschijnlijkheid is voor de daarop volgende letter etc.) Het voordeel van Huffman codes - en de reden dat deze zo populair zijn - is echter dat een Huffman code redelijk goed werkt op alle files die een zin of meer een regelmatig alfabet hebben, dus onafhankelijk van de gebruikte taal, zelfs andere code doet het voor nog vrij goed.

In mijn programma gebruik ik de bytes van de file als alfabet, de meest gebruikelijke en voor de hand liggende keuze. Dit geeft een alfabet van 256 karakters. Dit werkt prettig op tekst files, en dat soort files wordt namelijk vrijwel nooit meer dan de helft van de 256 karakters

gebruikt en dus werden bovendien ook nog niet allemaal even vaak gebruikt. In 125 moet een tabel voor de vertoelningen van de letters A-Z en de spatie in de originele taal. De spatie (10 bits) en de E (10.530) krijgen veel vaker voor te komen dan bijvoorbeeld de J (6.584) en de Z (4.0.081). Deze getallen liggen voor ander tabel vertaaler anders, maar in alle talen zijn er vrij grote verschillen tussen de verschillende letters.

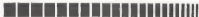
De mate waarin de file verkijnd wordt is vrij sterk afhankelijk van het soort file. Tekst files maken ongeveer 30% besparing aan file ruimte, door de vele spaties erin tot bijna 85% komen, machine code files maken ongeveer 50% (of afname betekent van 100% naar 100-50%). De enige soort files die ik heb kunnen ontdekken waarbij geen afname meer toename van de lengte optreedt waren de de files die het compact programma zelf aflevert, en zeer korte of lege files.

Uiteraard heeft het inspecten ook nadelen. De twee voorreente redenen komen voort uit het feit dat we een variabele lengte code hebben.

1- De Files zijn niet langer "random access". Dit komt erop neer dat we het 800ste karakter in de file te kunnen bereiken eerst de voorafgaand 175 afgesloten moeten worden.

2- Als er een buffer optreedt kan het daardoor ontbreken karakter een ander lengte hebben dan het originele karakter. De daaronder volgende karakters zullen dus ook fout gelezen worden, onder hun "begin punt" dan ook is.

Hoe maakt men een Huffman code? Eerst wordt van alle bytes bepaald hoe vaak ze voorkomen. Dan wordt de code geschrude. Hiermee worden de twee bytes die het meest voorkomen samengevoegd, ze vormen een binaire boom, waarbij de ene byte aan de linker en de andere aan de rechter tak hangt. Deze boom wordt gericht even vaak voor te komen als de twee bytes waarnaar hij is samengevoegd samen, en wordt hij de rest van bytes teruggezet. Dit proces wordt nu weder herhaald, waarbij ook de boom zelf gewogen met alle bytes mee doen, totdat er nog maar een boom over is. Deze houdt dan de code zelf in zich. Het een 0 of een 1 is aan te geven of we een linker of rechter tak moeten nemen op weg van de wortel van de boom naar een byte. Omdat de bytes die het meest vaak voorkomen het snelst gekompenseerd worden komen ze het laagst in de boom terecht, en de vaak voorkomende komen dicht bij de wortel terecht. Merk op dat bytes die niet in de file voorkomen gewoon niet meedoen, we hebben immers ook geen codering voor die bytes nodig. Figuur 1 geeft het compactie proces weer voor een file van 16 bytes.



INLEIDING

File... **reducedbbs.bbs**
header (150 bytes, type=0)

Frequentie tabel

char | freq char 2

a	5
b	6
c	3
d	8

versameling bomen

a	b	c	d
1	1	1	1
1	1	1	1
a	b	c	d

(code)meer 0 bomen

a	b	c
1	1	1
a	b	c

(code)meer weer bome

a	10
1	1
1	1
c	d

(code)meer weer bome

a	10
1	1
1	1
c	d

sundboom 00 a (c d33,b)
coderen als: 110'a 10'c'd'd'b'

Conversie tabel

code to code

a	00
b	1
c	010
d	011

Vertaling

a	00	0000 0000	:
a	00	0000 0000	:
b	1	0000 0000	:
a	00	0001 0000	:
c	010	0000 0000	:
c	000	0000 0000	:
a	00		:
d	001	1101 0000	:
b	1	1101 0100	:
b	1	0011 0000	:
c	010	1001 0000	:
d	011	1000 110	:
b	1		:
a	00	0	:
a	0	0001 0001	:
b	1	0010 0001	:
		1110 1001	:
		1100 11	:

LITVOOR

code

Gebruik van de programme's.

Er zijn twee manieren om de programme's te gebruiken. De eerste is om te gebruiken het EXEC of EXEC_2 aan te roepen. De computer heeft bij mij 'opti' en de programmeer 'unopt'. Als het programmeer geselecteerd is wordt in de rechter vensterhoek van het beeldscherm een klein windowje geopend. Er worden een invoer en een uitvoer file gevraagd, en als die correct ingevoerd zijn wordt het programmeer op de invoer file losgelaten waarbij zijn uitvoer op de uitvoer file terecht komt. Als er bij het ingeven van de filenamen fouten gemaakt worden dan komt er een GOOS foutmelding uit de EL terug. Dit is ook de voornaamste reden dat er in het Engels om een in- en uitvoer file gevraagd wordt, ik kan het niet uitlezen als een programmeer half Nederlanders en half Engels spreekt). Na een foutmelding geveed de goede filenamen intikken. Als er een verkeerde invoer file is opgegeven dan is het makkelijk om geveed op de EXEC toets te drukken als er een uitvoer file gevraagd wordt het file verveel begint dan weer van voren af aan. Nadat het programmeer klaar is het vertaalen wordt er gevraagd om een volgende in- en uitvoer gevraagd. Als men niet meer files te vertaalen heeft dan moet er bij de vraag om een invoer file de ENTER gedrukt worden. Het deze eerste methode kunnen dus meerdere files na elkaar verwerkt worden zonder de programme's te moeten herladen.

De tweede methode heeft gebruik van de 'org, test, bbslist'. Het aanroepen gaat dan volgens 'EX opt,myfile,myfile_opt', dus met de uitgebreide variabe van EXEC die in de toelichting uit, merk op dat opt, omdat de linker file twee keer geselecteerd wordt zijn invoer alleen van een file, en dus niet van een pipe. Dat kan dan... 'a,b,c' kan krijgen. Dus 'EX opt, myfile TO opt,myfile_opt' kan NIET, maar 'EX opt,myfile TO opt,myfile_opt' kan wel. Dropt een zijn invoer uit van ieder soort device, en dus ook pipe's, helten.

Het programmeer opt is in staat om twee foutmeldingen te genereren. 'bed parameter' als de invoer geen echte file is en 'buffer overflow' als er een code langer dan 30 bits zou ontstaan. Dit laatste blijkt in de praktijk vrijwel nooit voor te komen. Verder geven beide programme's overvloedige foutmeldingen die uit GOOS komen door een de generator. Omdat unopt niet 100% veilig is bij gebruik op files die niet eerst door opt gegeven zijn is het verstandig alle geaccepte files op 'opt' te laten eindigen en unopt alleen op die files te gebruiken.

Figuur 1.
Opt een het werk op een 10 byte file
(reductie tot 15 bytes)

Opmerkingen over de source files.

Zieke iedereen die veel vooruit heeft gegaan naar de listings zal zijn opgevallen in de zoek in machine code beschreven. Welke kant niet iedereen machine code op de GL, hetgeen erg jammer is want de GL is de machine code op zijn best. Voor de pure 68000 machine code heb ik gebruik gemaakt van een tabel in FBI, die een samenvatting geeft van de 68000 instructies en hun betekenis. Dit is echter heugst waarschijnlijk een wat oudere introductie voor iemand die nog nooit in een andere machine code heeft gewerkt. Er bestaan waarschijnlijk wel boeken waarin een redelijk goede buffer 68000 code wordt gegeven. Voor de GL specifieke zaken (trap 62, io, fpu, etc.) heb ik gebruik gemaakt van FBI, waarin alle GL trap 6 en sectoren beschreven staan.

Aan de listings is wel te zien, dat ik de source files regel naar een commentaar heb voorzien. Een soort van commentaar? Het is uiteraard niet nodig om al dat commentaar ook in te lezen als het alleen de bedoeling is om op te zoeken wat te krijgen. Een commentaar in een source file is altijd handig bij het foutzoeken, maar verder kan men natuurlijk ook gewoon dat tijdschrift lezen en dat als naslagwerk voor deze programme's te gebruiken. Het is heel erg nuttig om te weten op de verschillen tussen de door mij en door de maker te gebruiken assembler. Ik heb de Hecacom assembler gebruikt, welke enkele kleine afwijkingen van de standaard 68000 code heeft. Alleen het is er 68000 oprecht, die het mogelijk maakt om een register een naam te geven. Dit is natuurlijk zeer handig in stukken code waar veel registers ingevuld gebruikt worden, maar voor assembleers die dit niet kunnen zal men gewoon de verschillende registernummers moeten gebruiken. Verder hadt deze assembler de "BET" opdracht, die in andere assembleers anders kan heten maar meestal wel aanwezig is. BET is het in veel programmeertalen INCLUDE heeft het argument (een file) wordt in de code "involgen gevraagd". Als laatste is er nog de mogelijkheid van de Hecacom assembler om te onthouden of een label absoluut of pu relatief is. De label LABEL(PIC) kan dan nogmaals meer nodig, relatieve labels worden gewoon als LABEL gebruikt en het (PIC) wordt er door de assembler voor aangebracht. Niet alle assembleers kennen dit trucje, en sommigen zullen dus een foutmelding geven.

Over de machine code zelf zijn nog twee interessante opmerkingen te maken. Als eerste zal het de maker opvallen dat opzet in het als in opzet op sommige momenten het vijf registers gebruikt worden om in en uitvoer buffers te berekenen. Dit dient om de snelheid enigzins op te voeren, door de 68000 traps steeds voor een aantal bytes in plaats van per byte uit te voeren wordt de tijd die een trap moet worden berekend sterk vermindert. In alle gevallen heb ik buffers van 16 gebruikt. Dit is een op de 68000 gebaseerd hoeveelheid, ik heb geen idee wat de ideale omvang voor deze buffers is. Verschillende aan een met verschillende andere buffer omvang toe, echter, het bepalen van de optimale omvang is nogal veel werk, men moet het programma dan laten lopen op invoer files van verschillende lengtes en van verschillende soorten achtergrond gegevens. Bovendien is er geen enkele reden om een te nemen dat 16 en uitvoer buffer omvang groot zouden moeten zijn, noch dat op en output buffers van gelijke omvang nodig hebben. Het echter wel vaststaat is het feit dat buffers de snelheid flink ophoort een versie van op die zonder buffers werkt (d.w.z. de io, fpu en io, fpu omvang gebruiken) wat een heel langzaam. De tabel van de tijden en respectie resultaten van de hier afgedrukte versie staat in Figuur 2. Een tweede bijzonderheid van de code is de splitsing van de assembler files in een stuk dat de files opent (open.asm) en een stuk dat het eigenlijke werk doet (cpu.asm en output.asm). De achtergrond hiervan wordt ergens anders apart uitgelegd. (wordt vervolgd)

Literatuur verwijzingen.

- [1] Jonathan Rearden, Data Compression with Huffman Coding, Byte mei 1986, pag. 80
- [2] Andrew S. Tanenbaum, Structured computer organization, 2e editie, pag. 40, fig. 2-12 Prentice Hall, 2nd edition, 1989
- [3] Elektronische computing - 3 De 68000 komt er rondig - deel 2, pag. 77, tabel 3 Elektronisch, 1986
- [4] Adrien Dickers, GL addresseerbaar guide, Adrien Publishing, 1989

	Type	Size	Size_cpu	afname	T_cpu4	T_cpu8
Pagina 2	Basic	1484	1288	22.00%	7	4
	asm.c	6137	5889	27.40%	12	10
	cpu.asm	12288	7440	37.44%	12	12
	asm_source	24896	14864	64.17%	27	21
	io	960	960	1.73%	7	5
	io	7420	5896	21.16%	12	8
	io	21344	14264	29.12%	27	21



```
# Huffman code file compressor
# version 2.00    Oct 6, Philip H95a
# A.S.V. Byte enl '88a

        brs,a      start              job's start in support
        dc.l      0                    :
        ds.b      4444b               rest van job header
job_name ds.b      7                  :
        dc.b      'compress',0        :

# Include support routine, zorgt voor een correcte aanroep van process.

        get       'fdki_sup_asm'

# Definitie van enkele GDOS konstanten.

fo.fctng equ      0
fo.mctng equ      7
fo.psect equ      442
fo.headr equ      447

# definitie van waarden voor het 'tag' veld in het record

char_char equ      0      = 00 binair
char_node equ      1      = 01 binair
node_char equ      2      = 10 binair
node_node equ      3      = 11 binair
char_only equ      4
not_used equ      5

# start van werkbijke 'compress' routine
# alle registers behalve d0 blijven ongewijzigd, stack blijft ook
# ongewijzigd.
# INPUTS : a0 = channel id van invoer file
#          a1 = channel id van uitvoer file
#          a4 = pointer naar data space
# OUTPUTS: a0 = GDOS error code (niet bij succesvol verloop)

process     move.l    d1-d7,a0,a1,a4     bewaar alle registers behalve d0
           move.l    a0,inp_id(a4)      zet channel id's in variabelen
           move.l    a1,out_id(a4)
           move.l    a7,err_ptr(a4)     bewaar stack pointer voor noodgevallen

           move.w    #0,d0              via hele array 12 bytes!
           move.l    a4,a1
           clr.l     d1,d1+
           dbrs      #0,clr_loop

# Nu wordt de header verwerkt. Als het een 'gewone' data file is alleen
# de eerste 4 bytes (length & type byte & access byte). Lengte is nodig om
# later te kunnen bepalen hoeveel karakters er later ge-uncompress moeten
# worden, type byte om te zien of de header helemaal of slechts 4 bytes
# aanwezig zijn. Access byte is overbodig maar zit daar tussen. Een niet
# data-file krijgt s'm hele header mee, die gaat later een terugkeer op
# te voorkomen dat executeerbare files na uncompress niet executeerbaar meer
# zijn. De header gaat gewoon in de file, niet in de header van de ge-
# compacte file: een gecompacte executeerbare file is zelf niet direct
# te executen, slechts na gebruik van uncompress.

moveq      #header_length,d2           file header is nodig
lea        header(a4),a1
moveq      #0,header,d0
lea        err_ptr,d0                  channel id invoer nog in a0
moveq      #0,d1                       controleer of het wel een file is
```



```

      long      bad_per      alleen de eerste 8 bytes als
      moveq     $8,d0        data file i.e type 8
      lshl.b    hd.type%40
      leaq      @header_length,d2 andere hele header
      moveq     $8,header%40,a1
      lsr       header%40,a1
      eorq      eorq

      move.l    hd.len%8,d1   d1 is file lengte
      lshl.l    d1
      bne       exit_pr      IF lege file THEN stijg we al afloor

```

Initialiseer invoer buffer.

```

      moveq     $0,d0        d0 & a0 in gebruik voor invoer buffer

```

Tel de hoe vaak de verschillende karakters voorkomen. Na zeker nu de
frequentie tabel in het array.

```

pass_one bsr      r_index      FOR d1 := lengte DOWNTO 1 DO
      addq.l     $1,%a03        READI char 1
      subq.l     $1,d1         freq char 1:=freq char 1+1
      bsr       pass_one      END FOR d1

```

Tel hoeveel verschillende karakters er voorkomen, en maak het array in
orde 1 alles met freq = 0 worden lege nodes, alle andere tweede worden
vertaals van type 8 boons (tagchar_only, len%1). Maak op dat door het
wissen van het array alles behalve het freq nu nog op 0 staat.

```

node_cnt equ     d1          lequr geeft een register een naaapjet

      moveq     $0,node_cnt
      moveq     $255,d3
      move.l    a0,a0
      moveq     $not_used,d0
      lshl.b    $0?
      bsr       unused
      moveq     $char_only,a0
      addq.b     $1,len_byte%a0
      addq.w     $1,node_cnt
      moveq     $0,tag_byte%a0
      addq.l     $0,a0
      bsr       a0,node_cnt
      END FOR d3

```

d4 = aantal niet lege nodes

Volgende stap is om de nu verkregen verzameling boons te gaan verspreken
tot 1 boon. Dit gebeurt door steeds de twee boons met de laagste freq
te nemen en die dan tot 1 boon te combineren. We weten dat dus d4-1
keer doen, dan is er 1 over. De nieuwe boon krijgt steeds als freq de
som van zijn sub boons.

Een speciaal geval is als er al slechts 1 boon is. Alle karakters in de
file zijn dan gelijk. Dit wordt in een aparte routine afgehandeld.

```

      subq.w     $2,node_cnt
      btl       word_chr      speciaal geval : node_cnt=1

```

```

x_freq equ     d0          geef de registers bijnamen
y_freq equ     d1
x_pr equ     d2
y_pr equ     d3
x_addr equ     a0
y_addr equ     a1

```




```
move.l d0,e_addr
move.b d0,e_ptr
bit_cont move.b y_ptr+rightfs_addr1
        dir.b    len_bytes(e_addr1
        move.l   x_ptr+fs_addr1
$
        ENDD IF
        ENDD FOR d1
```

\$ Nu gaat de conversie label gemaakt worden. Hoewel de frequenties daarbij \$ overgeschreven worden blijft de vore van de boos bevraagd.

```
move.l   x_ptr,d0          d0 is vorel van de boos
move.l   d0,d7             bevaarsel
ber.a    bit_conv_table     maak de conversie tabel, vore boos nog
move.l   d7,d0             tabel? len & freq niet weer!
```

\$ Nu gaan we de boos naar de vinder schrijven. Vanaf hier zal de uitvoer \$ buffer actief zijn, d.w.z. ad, de en d7 mogen niet meer veranderd worden \$ door de code.

```
move.l   $1000000(out_len/40+13+01),d6      init out_buf
lea      out_buf+47,d6
ber      out_len                schrijft de boos
```

\$ Nu beginnen we aan PASS TWO over de invoer. Hierin wordt de invoer \$ gepast in de geconstrueerde code. Eerst moet een "read" op de invoer \$ gedaan worden.

```
moveq    $0,d1              we doen een "read(0)"
moveq     $16,pass,d0        "
move.l    inp_idia47,d0      "
ber       qd3                "
$
moveq     $0,d3              invoer buffer ook readen
```

\$ Maak nu PASS TWO.

```
pass_len move.l    hi,lania47,d1      FOR d1 in lengte_invoer DOWNTO 1 DO
ber       t_index              readByte1:
move.l    ia01,d6
moveq     $0,d3                -- zorg d3.L i.p.v qd3.B = len
move.b    len_bytesia01,d3
ber       send_bisstring        write_bis((readByte1, lenByte1)
subq.l    $1,d3                ENDD FOR d1
moveq     pass_len
```

\$ De uitvoer buffer moet worden weggeschreven, ook als hij niet geheel vol is. \$ Voor een beschrijving van de werking van de buffer zie de "buffer sectie".

```
flush_out move.w    out_len,d2      d2.w = buffer lengte in bytes
addq.w    $1,d6                  d6.w = aantal ongebruikte bits in d7
moveq     $0,d9                  d7 := d7 << d6.w
move.w    d6,d8                  "
sal.l     d0,d7                  "
move.l    d7,ia07+               bit buffer naar byte buffer.
move.w    $0,d6                  d6 := aantal ongebruikte bytes in d7
sub.w     d9,d3                  d3.w = buf lengte - ongebr. in d7
moveq     d6                      d6.w = ongebruikte lang worde in buf-1
sal.w     $0,d6                  maak er bytes van
sub.w     d9,d3                  d3.w = aantal gebruikte bytes
beq.s     to_exit                IF d3.w=0 THEN
ber       flush_out              schrijft buffer uit
$
        ENDD IF
to_exit   moveq     $0,d0          geen error
ber       exit_ptr               EXIT proces
```




```
% INPUTS : d0 = variabele
% OUTPUTS: geen
% COMMENT d0,d3,d4 en a0, buffer gebruikt d0,d7 en a0.
% RECURSIEF :
% NB: bose begint altijd met een knoop

sub_rine bose_a      send_1      knoop d0 -> send "1"
bose_a      send      a0 := d00
move_b      leftia0,d0      write_sub_bose(left)
blat        00,flag_byte(a0)
bose_a      sub_rl
move_b      rightia0,d0      write_sub_bose(right)
blat        00,flag_byte(a0)

% Subroutine om subbose weg te schrijven.
% INPUTS : d0 = left of right subbose.
%      WORD flag : set = blad, reset = knoop
% OUTPUTS: geen
% COMMENT d0,d3 en d4, buffer gebruikt d0,d7 en a0.
% RECURSIEF :
% N.B. a0 wordt NIET veranderd.

sub_rl      seq_0      send_0_byte      IF blat THEN
%      aave_1      a0,-1eq1      write "0" karakter_code
bose_a      seq_1ree      ELSE
move_1      cap0+,a0      write "1" left_sub_bose right_sub_bose
%      blat
%      seq_1

% EINDE high-level subroutines.
% =====

% Geval alle karakters in file gelijk. We nemen nu alleen de bose uit te
% schrijven, die nu uit 1 blad bestaat. Ongeacht hoe de oorspronkelijke
% file nu terugrijgen door de code is dat ene blad "lengte" keer te
% herhalen. Verschil met het "gewone" geval altijd te achterhalen. Hier
% bestaat de bose uit 1 blad en begint dus met een 0 bit, terwijl normaal
% de bose met een knoop begint (lees anders minmaal twee karakter codes), en
% het eerste bit dus een 1 is. Merk op dat het geval van 0 verschillende
% karakter codes altijd direct aan de header bekend kan worden: lengte 0"

aave_chr aave_1      a00000000out_1eq/4)=1+01,d0      init output buffer
%      blat      out_buffer(a0)

bose_a      seq_1      schrijft "0" karakter_code
bose_a      flush_ext      flush buffer byte 1 en uit

% Subroutine voor aave_chr. Deze routine zoekt de enige karakter code die
% in de file voorkomt, en dus als enige een 1 in zijn 1eq_byte heeft staan
% en produceert dan i"0" karakter code) op de uitvoer.

% COMMENT d0,d3,d4,a0: buffer gebruikt d0,d7 en a0.

sch_1      aave_1      00,d0      d0 := 0
move_1      a0,a0      a0 := d00
sch_loop  lat_0      1eq_byte(a0)      WHILE len(d0)=FALSE DO
bose_a      aave_0_byte      BEGIN
addq_a      00,d0      d0 := d0 + 1
addq_1      00,a0      a0 := a00
bose_a      sch_loop      END
%      EXIT via aave_0_byte.
```



```
* subroutines (low-level).
* =====
* Subroutine die een "0" en de kode is 00.0 naar de uitvoer stuurt. De
* kode in 00 komt uit het minst significante bit eerst naar buiten!
* INPUTS : 00.0 = kode
* OUTPUTS : geen
* GEBRUIKT registers d3 en d4, buffer gebruikt d6,d7 en a6
```

```
send_0_byte
    moveq    $0,d4          send "0"
    bsr.s    send_bit       .
    moveq    $0,d4          send_bits($0,4)
    moveq    $7,d3          .
```

```
* Routine die een bitstring met lengte 1 tot 32 kan schrijven naar de uitvoer.
* INPUTS : d4 = te verzenden bits in meest-rechts bits, d3 = lengte = 1.
* OUTPUTS: geen
* GEBRUIKT d3 en d4, buffer gebruikt d6,d7 en a6.
```

```
send_bitstring
    bsr.s    send_bit       .
    dbrs    d3,send_bitstring
    rts
```

```
* Routine die "1" naar de uitvoer schrijft.
* GEBRUIKT d4. Buffer gebruikt d6,d7 en a6.
```

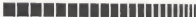
```
send_1      moveq    $1,d4
```

```
* Routine die het meest-rechts (=LSB) van d4.1 naar de uitvoer schrijft.
* d4 := d4 DIV 2 = d4 >> 2.
* Buffer gebruikt d6, d7 en a6.
```

```
send_bit     moveq    $1,d4          Mees LSB van d4
send_n       moveq    $1,d7          Stop die in d7 0-bit buffer)
            dbrs    d6,bit_rts      IF d7 vol THEN
            swap    d6              Stop d7.1 in byte buffer
            moveq    d7,byte_rts     .
            dbrs    d6,buf_rts      IF byte buffer vol THEN
            moveq    d6-d3/d0-a1,-(sp) schrijft buffer weg
            moveq    Rout_leng,d3   (4 error संगदत दल गेन
            bsr.s    flush_out      registrars अतुरलल रहलन 4)
            moveq    (sp)+,d6-d3/d0-a1 .
buf_rts      swap    d6              END IF :
            moveq    #0,d6          d7 is voor lang
bit_rts      rts                  END IF :
```

```
* Routine die een byte inleest en de aangesprokene routine het adres in
* het array geeft dat daarbij hoort. De invoer is gebufferd.
* INPUTS : geen.
* OUTPUTS: a6.
* GEBRUIKT d6, buffer gebruikt d3 en a6.
```

```
f_index      dbrs    d3,in_buffer    IF buffer_empty THEN
            moveq    d6-d3/d0-a1,-(sp) (4 bewaar registers ti
            moveq    #ing_leng,d2     lees een buffer vol
            addq     d3,d3            en herstel a6 & d3
            moveq    (sp)+,d4-a1,a6   .
            lsr     (sp,buf_leng),a6 .
            moveq    a6,a1           .
            moveq    #0,buf_leng,d6 .
            bsr.s    qd3            .
            ldr.s    d1              IF niete gelezen THEN
            moveq    err_rts        RAISE end_of_file
            moveq    (sp)+,d6-d3/d0-a1 END IF (4 herstel registers ti
            END IF
```



```
in_buffer
move.b    (a0),d0          move d0 uit buffer
index     a0!,a    a1!,d0    d0,a := d0,b
          a1!,a    d2,d0      a0 := a++d0d0,e
          a0,a1     a4,a0      (a := d0d1) !
          adda.w    d0,a0      ,
          rta

% Routine die de uitvoer buffer naar de uitvoer file schrijft.
% INPUTS : d0,a = hoeveel bytes er in de buffer zitten.
% OUTPUTS: d0,e = ruimte voor hoeveel lang woorden in buffer -1.
%      a0 = eerste adres uitvoer buffer
% GEWISHT d0,d1,d2,e0,a1 id=0,d3=-1 na deze routine!

flush_out
move.w    #((out_buf/4)-1),d0  meet d0,e & a1
lea       out_buf(a0),a0

move.l    a0,a1              EXIT via a0

% Routine die een lg. string uitvoert a0:
% - a0 = id uitvoer file.
% - time_out = oneindig.
% INPUTS : d2 = lengte te schrijven.
%      a1 = adres uit te schrijven buffer.
% OUTPUTS: a1 = adres voorbij buffer.

% GEWISHT d0,d1,d3 en e0.

string
move.l    out_id(a0),a0        channel id
move.w    d0,d1,d3            EXIT via a0

% Subroutine die een TRAP #3 naar glos doet a0:
% - time_out = oneindig
% - Error return naar aanroeper van proces voor alle errors behalve end of file.
% PARAMETERS als trap #3, d3 altijd -1, error 0 of err.ef

qd3
move.w    #1,d3              time_out := -1 oneindig
trap      a0
lea.b     d0
beg.s     qd3_rta            IF no error OR error end of file THEN
                                RETURN
copy.b    err.ef,d0          ELSE
beg.s     qd3_rta

% Routine die de proces routine afbreekt en terugkeert naar de aanroeper
% ervan. Er moet een error code in d0 zitten.
% Entry err_rta bruikbaar van uit elke subroutine! dus voor een error.
% Entry exit_pri alleen bruikbaar via brs of jump vanaf buitenste
% subroutine nive. dus voor een "normale" exit

err_rta
move.l    err_mick(a0),a7      Stack herstellen als of we in proces zitten.
exit_pri
move.l    a7,a0,d0            (a7+1,d1-d7/a0-a6) Registers herstellen
rta

% Reutel routines die een exit met een bepaalde fout veroorzaken

err.ef
eqa       -10
err.be
eqa       -5
% err.bp import van support

bad_par
moveq     err_bp,d0           error ! bad parameter
bra.s     err_rta

buf_full
moveq     err_be,d0           error ! buffer full
bra.s     err_rta
```



```
# record_def offset  
# eerste 4 bytes lang word freq/code -> geen offset
```

```
left    equ 4  
right   equ 8  
freq_byte equ 6  
len_byte equ 7
```

```
# array [0..255] of record  
# begin direct op 1a40
```

```
array_size equ 2560
```

```
# Er kunnen nooit meer dan 256 records nodig zijn! alle karakters hebben  
# iedere code in het gebied 0..255 en zelfs een volledig volle boom  
# bestaat maximaal  $2^7+2^6+...+2^0 = 255$  codes
```

```
# gebruik:  
# Het array doorloopt drie statia: frequentie tabel, versameling boom en  
# conversie tabel.
```

FREQUENTIE TABEL

```
# Slechts de eerste vier bytes van het record zijn in gebruik. Hierin zit  
# een lang word met het aantal malen dat het karakter zwaer code de index  
# van dit record heeft
```

VERSAMELING BOOMEN

```
# Het array wordt nu als een verzameling 'nodes' beschouwd. Een node is de  
# wortel van een boom als zijn len byte < 0 is. Het freq veld bevat dan  
# het aantal malen dat een karakter uit de boom in de invoer file voorkomt.  
# dat de een van de individuele frequenties van die karakters. Er zijn  
# drie soorten nodes: laag, type A en type B.  
# Laag nodes hebben de code 'not_used' in hun len veld. Ze kunnen nooit  
# de wortel van een boom zijn dus het len veld is 0. Ander velden worden  
# niet gebruikt.  
# Type A hebben de code 'char_only' in hun len veld. Ze zijn altijd in  
# hun eenzige een lokale boom, dus het len veld = 1. Zodra ze een een  
# andere node tot een boom verzamelt worden worden ze of laag of type B. De  
# left en right velden zijn ongebruikt.  
# Type B nodes hebben in hun len veld de codes 0..3. Binair zijn dit de  
# codes 00,00,10 en 11. Bit 0 geeft aan of het right veld een 'char' of  
# een 'node' is, bit 1 doet dit voor het left veld. Een 'char' is een blad  
# aan de boom, de karakter code staat direct in het left resp. right veld.  
# voor een 'node' staat hierin echter het index nummer van een volgende  
# node aan het veld.
```

CONVERSIE TABEL

```
# Het array bevat nu voor iedere karakter code een variabele lengte code.  
# De lengte van die code staat nu in het len veld, terwijl de code zelf  
# in freq veld staat, eerste bit in het len, in de left, right en len  
# velden blijft de vorm van de boom bewaard.
```

```
# Buffers:  
# =====
```

```
inp_len equ 1024          lengte buffers (op de gski)  
out_len equ 1024          (eest 4-word zijn!)  
inp_buf equ array_size    start adressen (rel. 40)  
out_buf equ inp_buf+inp_len
```



```
* Input buffer:
* De input buffer bestaat uit inp_len bytes vanaf inp_buff+1. In dit werdt
* bijgehouden hoeveel bytes er nog in de buffer zitten. ad wijst naar het
* eerst volgende byte dat gelezen moet worden. De lees_een_byte routine
* werkt als volgt:
* - indien de buffer leeg is -> lees en sla een buffer vol is:
* - haal een byte uit de buffer.
* De buffer kan nooit helemaal vol zitten: er wordt voorzichtig na het
* inlezen een byte uitgehaald.
* Initialiseren gaat dus als volgt: ds:=0. de buffer bevat niets meer.
```

```
* Output buffer:
* De output buffer bestaat uit dt en out_len bytes vanaf out_buff+1. Er
* worden bits gebufferd in dt, dat bevat het aantal bits dat nog vol is
* in dt -1. Als de andere helft van dt worden voor een long word buffer
* gebruikt. Als wijst naar het eerst volgende vrije long word, en is het
* bovenste word van de stack het aantal vrije long word? plaatsen in die
* buffer -1. De bits in dt schrijven van rechts naar links.
* Werking lees_een_long routine:
* - schuif bit in dt.10
* - als dt.1 vol dan gaat hij naar long word buffer:
* - als die daardoor volraakt dan gaat hij naar de volgende file.
* Zowel dt.1 als de long word buffer kunnen nooit vol zijn, zodra ze dat
* zijn worden ze leers direct doorgegeven.
* Initialiseren gaat als volgt: ds := (bovenste helft: out_len/2-1,
*                      onderste helft: 0)
*                      ds := 'outbuff+1'
* Na gebruik moeten zowel dt als de buffer geleegd worden. i.e. hun inhoud
* moet naar de uitvoer file geschreven worden.
```

```
* Variabelen:
* =====
```

```
inp_id      equ      out_buff-out_len      (long) Channel_id inder file
out_id      equ      inp_id+1              (long) Channel_id uitder file

err_stack   equ      out_id+1              (long) positie in stack voor een return
                                                "overal doors doorheen".

header      equ      err_stack+4           114 bytes: opslag voor de header van
*                                                de inder file.

* Header fields:
hd_len      equ      header                (long)   lengte van de file
hd_addr     equ      header+4              (byte)   access byte, nut onbekend
hd_type     equ      header+5              (byte)   type : 0 normaal, 1 wsec, 2 relac
hd_attr     equ      header+6              18 bytes:extra info, type afhankelijk,

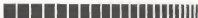
header_length      equ      14             lengte van de header is 14 bytes.,
```

```
* geheugen gebruik:
```

```
* array      -> 2k      256 k B bytes
* buffers    -> 2k      2 k B bytes
* variables  -> 12      3 long words
* header     -> 14      14B file header
* stack      -> 312     reserve:we routines hebben veel stack nodig
* support    -> 0        Single support gebruikt geen data geheugen.
* -----+
* TOTAL      4644
```

size 4644 Data space | sommige assemblers willen DATA 4644

end



spectrum

EXPLORER 22-23

Johannes Ad Pasterkamp

Tegenwoordig zijn de meeste 28-SPECTRUM gebruikers in het bezit van een microdrive, een opslagmedium dat vele malen sneller werkt dan de conventionele tape-reclorder. Deze microdrive werkt met cartridges, een huidige tape waarop zo'n één- of twee programma's kan worden opgeslagen. De inhoud van een cartridge kan worden opgespoord door een DAT (data) opdracht. Op het beeldscherm verschijnt dan de programma's. Het nadeel dat hiervan voortvloeit is, is dat we aan deze catalogus niet kunnen zien of het om een BASIC-programma, machinencode of precompiler gaat. Ook kunnen we niet zien in welke regel een BASIC-programma start en wat de startadressen en lengte van machinencode programma's zijn. Het volgende programma buikt hiervoor een passende oplossing. Met behulp van het programma "EXPLORER" is het mogelijk om alle gegevens op een schijfje af te drukken. In de opgegeven catalogus kunnen we ook zien in welke regel een BASIC-programma start en wat het startadres is van een machinencode programma. Ook wordt de lengte in bytes aangegeven. Sommige programma's die zijn gevraagd met behulp van GOTO's zijn normaal niet zichtbaar in een catalogus. Onderstaand programma maakt ook deze programma's zichtbaar. Het programma bestaat uit twee delen. Eerst moet u het loader-programma intypen, waarna deze zichzelf aan het laden neemt. Daarna kunt u het tweede programma in. Dit programma kunt u geven met de opdracht: BASIC >GOTO "EXPLORER" LINE 3. Wanneer u een BASIC als machinencode op een cartridge plaatst, dan kan het programma weer worden geladen d.w.v. LOAD >GOTO "EXPLORER". Wanneer het hoop is ook de "CPUS DISCOVERY"-versie van dit programma klaar te hebben, die dan beschikbaar ook gepubliceerd zal worden.

Loader-programma's

```
10 FOR machinencode loader
20 PRINT "MACHINENCODE LAADPROG"
DATA%, "1" PRINT
30 FOR x=44000 TO 44220
40 READ code
50 FOR y=code
```

```
60 NEXT y
"1" PRINT "Plaats nu een cartridge
op in" PRINT "microdrive" : see
druk op ENTER."
```

```
80 PRINT "Met machinencode prog
laad de cartridge."
```

```
90 PRINT AT 25,0: FLASH 1:GOTO
1: daarna op een teletele: PRINT 0
90 GOTO 44000: EXPLORER CODE
44000, 330
```

```
100 DATA 207, 49, 42, 181, 92, 237
110 DATA 96, 99, 92, 147, 237, 82
120 DATA 125, 254, 5, 54, 3, 24
130 DATA 8, 12, 121, 4, 34
140 DATA 237, 92, 207, 84, 217, 220
150 DATA 207, 42, 3, 205, 1, 22
160 DATA 205, 152, 24, 237, 47, 214
170 DATA 92, 33, 205, 205, 34, 205
180 DATA 92, 33, 54, 254, 34, 237
190 DATA 92, 207, 54, 233, 225, 205
200 DATA 109, 4, 205, 232, 33, 233
210 DATA 124, 25, 205, 247, 25, 4
220 DATA 202, 4, 237, 47, 204, 92
230 DATA 205, 194, 18, 205, 20, 20
240 DATA 12, 204, 231, 124, 47, 233
250 DATA 102, 74, 254, 3, 2, 5
260 DATA 205, 205, 18, 24, 118
270 DATA 231, 124, 47, 181, 75, 118
280 DATA 33, 11, 205, 17, 12, 8
290 DATA 4, 4, 231, 74, 11, 123
300 DATA 182, 44, 205, 225, 221, 220
310 DATA 107, 4, 18, 33, 35, 124
320 DATA 221, 109, 71, 32, 5, 35
330 DATA 221, 33, 14, 245, 192, 221
340 DATA 225, 225, 44, 74, 44, 4
350 DATA 25, 13, 24, 123, 227, 212
360 DATA 107, 227, 247, 33, 205, 33
370 DATA 94, 205, 202, 12, 205, 14
380 DATA 9, 44, 77, 225, 42, 9
390 DATA 232, 20, 232, 237, 184, 192
400 DATA 225, 225, 221, 227, 221, 12
```

a

```
410 DATA 47, 119, 35, 221, 124, 44
420 DATA 119, 25, 4, 18, 221, 124
430 DATA 71, 119, 221, 35, 74, 54
440 DATA 247, 221, 225, 221, 124, 12
450 DATA 44, 221, 119, 12, 254, 54
460 DATA 44, 4, 205, 18, 19, 194
470 DATA 77, 205, 174, 205, 247, 22
480 DATA 221, 225, 225, 17, 44, 4
490 DATA 25, 205, 84, 29, 205, 174
500 DATA 201, 33, 11, 202, 17, 12
510 DATA 4, 221, 74, 12, 225, 182
520 DATA 102, 179, 221, 221, 225, 22
```

b

```
530 DATA 213, 107, 205, 84, 32, 18
540 DATA 205, 119, 221, 42, 245, 205
550 DATA 182, 29, 24, 184, 229, 35
560 DATA 124, 221, 119, 12, 205, 118
570 DATA 224, 225, 245, 175, 205, 24
```

c

```
580 DATA 23, 249, 32, 95, 205, 192
590 DATA 224, 221, 124, 82, 254, 8
600 DATA 44, 12, 254, 3, 48, 54
610 DATA 42, 225, 205, 182, 29, 221
620 DATA 124, 87, 205, 119, 245, 220
630 DATA 21, 174, 44, 205, 182, 27
640 DATA 244, 44, 27, 42, 34, 205
650 DATA 182, 29, 24, 54, 221, 240
660 DATA 94, 124, 32, 44, 42, 202
670 DATA 205, 182, 29, 221, 74, 89
680 DATA 221, 74, 94, 205, 184, 254
690 DATA 24, 20, 42, 175, 205, 182
700 DATA 29, 221, 74, 82, 221, 74
710 DATA 84, 205, 182, 221, 42, 44
```




```

200 DATA 98,248,32,3,13,32,248,
13,803
220 DATA 22,1,129,130,254,170,4
8,13,767
240 DATA 71,209,44,135,180,40,5
,87,815
260 DATA 187,32,1,197,130,147,7
1,213,840
280 DATA 187,120,47,128,7,43,13
4,82,885
300 DATA 130,100,126,245,205,17
0,34,71,1588
320 DATA 136,167,32,3,241,120,4
93,131,1588
340 DATA 248,7,79,24,11,4,82,1,
434
360 DATA 15,16,252,193,180,180,
119,193,1232
380 DATA 236,12,40,8,200,88,255
,40,832
400 DATA 181,140,130,254,254,32
,140,123,1329
420 DATA 237,88,98,254,1,200,19
3,5,1039
440 DATA 34,184,197,82,127,219,
254,31,1048
460 DATA 210,0,13,285,170,34,71
,4,707
480 DATA 128,7,14,253,230,1,193
,181,1007
500 DATA 0,0,88,8,8,88,8,136
520 DATA 17,0,88,0,17,0,88,0,17
0
540 DATA 85,0,82,0,88,0,88,0,34
0
560 DATA 85,170,82,170,88,170,8
8,170,1829
580 DATA 8,8,255,0,0,0,288,0,83
0
600 DATA 88,88,88,88,88,88,88,8
8,844
620 DATA 8,255,0,288,0,288,0,13
5,1020
640 DATA 170,170,170,170,170,17
0,170,170,1380
660 DATA 88,88,288,88,88,88,255
,88,838
680 DATA 288,183,193,255,255,15
3,243,288,1838
700 DATA 17,34,48,138,17,34,48,
138,830
720 DATA 138,48,34,17,138,48,34
,17,830
740 DATA 183,81,123,234,183,81,
123,234,1030
760 DATA 183,204,193,81,183,204
,193,81,1030
780 DATA 1,130,88,40,14,82,88,1
35,837
800 DATA 138,85,34,85,138,85,34
,85,880
820 DATA 98,144,0,4,98,144,0,4,
510
840 DATA 34,17,17,34,48,138,138
,88,810
860 DATA 88,238,110,34,88,238,1
10,34,818
880 DATA 38,2,2,180,144,144,254
,18,785
900 DATA 32,32,32,255,2,2,2,255
,812

```

HYPERTEXT

HyperText is een zeer verrassend programma. Het verslaapt niet wat het doet. Probeer zelf maar.

De het te laten runnen, typ je eerst de aanpaktaal in, door middel van het eerste programma. Je zet de code op cassette of drive met 1 NAME "name" CODE 33798,118 (al dan niet met Prg"11).

Als tweede typ je het BASIC programma in, en zet je het op hard-drive met het toegewezen LINE 7000, zodat het programma bij ingaven automatisch op die regel opstart.

Bedien je met cassette werkt, dien je de syntax van de name en load-command's natuurlijk aan te passen. OK.vdm.1

```

10 LET B=100
20 FOR A=23284 TO 33413 STEP 8
30 LET B=C
40 FOR B=A TO A+7
50 READ C POKE B,C LET B=B+D
60 NEXT B
70 READ C IF C=0 THEN PRINT
"FOOT IN ROOM",B
80 LET B=B+10 NEXT A
100 DATA 17,7,248,123,188,48,2,
98,781
120 DATA 87,74,205,149,91,47,85
,183,790
130 DATA 78,225,100,91,31,83,11
1,123,829
140 DATA 148,98,32,3,124,145,10
3,238,897
160 DATA 28,23,17,7,144,123,18
8,48,790
170 DATA 2,90,87,123,148,60,245
,67,820
180 DATA 14,0,288,170,34,243,18
3,88,845
190 DATA 23,0,18,288,71,197,229
,42,818
200 DATA 123,34,8,120,254,1,40,
255,845
210 DATA 32,1,123,174,188,38,14
,243,743
220 DATA 225,38,124,238,7,38,9,
37,780
230 DATA 1,32,0,0,124,228,240,1
83,747
240 DATA 183,18,218,201,121,120
,288,18,1240
250 DATA 18,18,111,121,228,7,40
,71,830
260 DATA 178,31,98,14,252,204,0
0,730

```

```

10 LET Y1=0: LET Y1=X1: LET X1
-255: LET Y2=175
20 INPUT "Berekeningstijl? ",d
30 IF d=1 OR d=2 THEN GO TO
25
31 CLS
32 PRINT 20;"DECLARE - druk op
SPACE"
40 LET X1=X1+5: IF X1>255 THEN
LET X1=X1-255
50 LET Y2=Y2-5: IF Y2<0 THEN
LET Y2=Y2+175
60 LET X2=X2-5: IF X2<0 THEN
LET X2=X2+255
70 LET Y1=Y1+5: IF Y1>175 THEN
LET Y1=Y1-175
80 POKE 23217,d: POKE 23228,x
2
90 POKE 23231,y: POKE 23232,y
3
100 RANDOMIZE USR 23244
110 IF (INKEY$="" ) THEN GO TO
40
120 GO TO 10
1300 LOAD "P",1:"BLOCK:xxx"CODE
23244-118: GOTO 24: 100

```

Van de heer Duglen uit Rotterdam, ontvang de redactie een cartridge met enkele kleine maar handige programma's.

Het eerste programma is WEDEN.

Dit is een tellprogramma dat veel voor de vrouw of vriendin een bepaald bereikpakken aan te geven van haar maak.

Wijzende van het oorspronkelijke patroon en het daarbij op te zetten aantal steken kunt u voor haar het aantal steken berekenen met overeenkomst met haar maak.

Voor elk onderdeel van het te breien stuk worden u de oorspronkelijke gegevens gevraagd en de Spectrum rekent dan het nieuwe aantal steken uit.

Het tweede programma is 10A.

Met dit programma kunt u de particuliere statistieken berekenen. De vragen binnen het programma spreken voor zich.

De rekenen is inmiddels naar de heer Duglen overgave.

INPDE 2 10A

```

10 CLS
20 PRINT "DIT PROGRAMMA BEREKE
MT VOOR U BESCHRIJVING VAN HET P
ATROON VOOR ANDERE OMSCHRIJVING"
30 PRINT "EERSTE VRAAG IS HOEVE
DEL STREKEN HET PATROON OMSCHRIJV
" "X" HEEFT.

```

```

10A HOEVEEL U GEGEVE
HERVOLGENDE HEEFT OMGE
HET PATROON U HEEFT U
10A DE HOEVEEL STREKEN DAT OMGE
OMGEHEERDE
40 PRINT

```

```

50 INPUT "hoeveel steken origi
neel" :a
60 PRINT "oorspronkelijk aantal" :a
70 INPUT "hoeveel steken gema
akt" :b
80 PRINT "gemeent aantal" :b
90 INPUT "welk onderdeel wilt
u weten" :c
100 PRINT "hoeveel steken " :a
110 "oorspronkelijk"
120 INPUT d
130 PRINT "oorspronkelijk aantal " :a
140 "a"
150 LET c=a/b
160 LET a=INT 10000
170 PRINT "aantal steken " :a/b
" wort"
180 PRINT a
190 PRINT "
-----
"
200 PRINT "Mag een berekening
?"
210 PRINT "Stoppen
?"
220 GO TO 240
230 IF INKEY$="1" THEN CLS : GO
TO 20
230 IF INKEY$="2" THEN GO TO 25
0
230 GO TO 200
240 LET a=INKEY$
250 RETURN
260 CLS
270 PRINT AT 12,12;"EINDE"

```

DE 2 DECTREKENTROEN

DE 2 DECTREKENTROEN
B. H. DUGLEN

```

30 CLS
40 LET a=0
50 LET y=0
60 LET x=0
70 POKE 23228,x
80 CLS
90 INPUT "welke rekentruut invoer
en " :a
100 LET b="123"
110 PRINT INK 7; PAPER 21a
120 PRINT

```



```

130 LET bedragnd
140 LET terugnd
150 INPUT "Hoeveel posten "pa
160 FOR a=1 TO a
170 PRINT
180 PRINT "Welk bedrag post "pa
190 INPUT a
200 PRINT
210 PRINT "post "pa;"="; PAPER
220 GOTO 130
230 PRINT
240 PRINT "Hoeveel terug "pa;"=";
250 INPUT terug
260 PRINT
270 PRINT "terug "pa;"="; PAPER
280 GOTO 130
290 LET bedrag=bedragnd+a
300 NEXT a
310 LET omv-bedrag
320 LET lynt-bedrag
330 PRINT
340 CLR
350 PRINT "totaal "pa;"="; PAPER
360 PRINT "totaal "pa;"="; PAPER
370 LET rest=bedrag-terug
380 LET omv-rest
390 PRINT "rest betaalde;" PAPER
400 GOTO 130
410 INPUT "Wilt u dit op printe
420 IF v=1 THEN GOTO 430
430 IF v=2 THEN GOTO 440
440 PRINT "totaal "pa;"="; bed
450 PRINT "totaal "pa;"="; ter
460 PRINT "zelf betaalde;"rest
470 PAUSE 50
480 GOTO 400
490 CLR
500 PRINT "Totaal rek,=";rk
510 PRINT "Totaal zelf bet,=";z
520 PRINT AT 21,2;"Wilt u dit o
530 INPUT rk
540 IF rk=1 OR rk=2 THEN GOTO
550
560 IF rk=1 OR rk=2 THEN GOTO
570
580 PRINT "Totaal rek,=";rk
590 PRINT "Totaal terug,=";tr
600 PRINT "Totaal zelf bet,=";z
610 PAUSE 50
620 GOTO 400
630 REM *****
640 BEEP .5,20 BEEP .5,20
650 CLR
660 PRINT AT 2,2;"nog een berek
670 PRINT AT 11,2;"stoppen
680 PRINT AT 13,2;"totaal over
690 PRINT AT 13,2;"

```

```

710 IF INKEY=" " THEN GOTO 720
720 IF INKEY="1" THEN GOTO 730
730 IF INKEY="2" THEN GOTO 740
740 IF INKEY="3" THEN GOTO 750
750 IF INKEY="4" OR INKEY="5"
760 GOTO 710
770 STOP
780 CLR
790 PRINT AT 13,13;"Einde"

```



WILDE-ABONNEE ??

Wat is een WILDE-ABONNEE ?

Een wilde-abonnee is een abonnee van de Sinclair Side, die voor 31 december 1984 besluit een abonnement op de Sinclair Side te nemen.

Doch deze antwoorden Sinclair gebruikers en veel vertrokken hebben in een tijdschrift, wil de redactie deze abonnees apart behandelen.

Alle beloningen zullen zij in de toekomst, volgens zijn abonnees zijn, steeds kunnen profiteren van kortingen op de door de Sinclair Side aangeboden artikelen.

Dit geldt ook voor de verspreidingskosten - cartridges met de programma's uit de Sinclair Side, vanaf nummer 5 worden er ook andere artikelen aangeboden.

Als dat een goede reden is om een WILDE-ABONNEE te worden....

De laatste dag is en blijft 31 december 1984, daarna zullen alle aanmeldingen als gewone abonnees worden aangemerkt.

Dus.. zorg dat je er bij hoort !

TASWORD 2 IN KLEUREN

Mit een vorig artikel in dit tijdschrift hadden we al geleerd dat de Tasword 2 regel vlot aan onze eisen kunnen aanpassen, met het uitbreiden van teksten betreft.

Er is nog een terrein waar Tasword zich opeenvolgend tot wijzigingen: de kleur van het scherm. Misschien lijkt dat element niet zo essentieel, maar het zijn de kleine dingen die het leven aangenaam kunnen maken. Wanneer je je tekstverwerker wilt aan een stuk gebruiken, kan het een heel stuk prettiger werken indien je bestuurscherm met kleurcombinaties versiert die jou aangenaam voorkomt.

Je kan de kleuren die Tasword gebruikt, op diverse manieren aanpassen. Het scherm is 32 karakters per regel-wide, of in 64 karakters per regel-wide, de kleur van de linker marge die je instelt, de kleur van de rechter marge, en de kleur van de onderste twee regels op het scherm, dat kan allemaal gewijzigd worden.

Eerst bekijken we de 64-karakter-wide.

De kleuren voor het scherm in de 64-wide kunnen we instellen door middel van vier POKE-instructies. Voordat moet je de gewenste kleurcode berekenen door 8 maal de kleurwaarde van de achtergrond (de PAPER-kleur) bij de kleurwaarde van de voorgrond (INK-kleur) op te tellen. We stellen de variabele K49 gelijk aan de berekende waarde.

Wij geven nu achteromvolgens de vier POKE-instructies:

```
POKE 56512, K4
POKE 56513, K44
POKE 56521, K4
POKE 56522, K44
```

en de nieuwe kleuren staan ingesteld.

De kleur van de marge.

Om de linker marge een andere kleur te geven dan de achtergrondkleur, moeten we een speciale berekening uitvoeren als voor de kleur van het 64-karakter-scherm. Denk eraan, dat de marge in papierkleur worden ingevuld. Na de berekening moet je de volgende twee instructies:

```
POKE 56508, K4
POKE 56509, (kleurcode)
```

Om de kleur van de rechter in 64-wide.

Om de kleur van de rechter in 64-wide te wijzigen, volstaat 1 enkele instructie. Je moet beide berekenen, eerst de code van de gewenste kleur in de instructie invullen:

```
POKE 48514, (kleur)
```

Om de kleuren van de 32-karakter-wide kunnen we veranderen.

De kleuren van het scherm in 32-wide kunnen eenvoudig bepaald worden door de gewenste BASIC-commando's waarmee je de kleuren instelt, evenwel in het programma zelf opgenomen. In deze code is evenwel de keuze beperkt tot de onderen kleur-codes, en de marge wordt automatisch ingesteld op de kleur die als code 1 onder heeft dan de kleur van het papier. Indien je de kleuren wijzigen in het programma draait, bijv. om te kijken dan wanneer je naar BASIC terugkeert.

Om de kleur van de rechter in 32-wide te wijzigen, vervang je de code van de gewenste kleur in de volgende instructie:

```
POKE 56546, (kleur)
```

De kleurinformatie voor de onderste twee regels worden op andere adressen opgeslagen.

Wende regels van je wijzigen door de kleurcode te berekenen zoals in de 64-wide en die code te gebruiken in de instructie:

```
POKE 44570, (kleurcode)
```

Voor de berekening van de twee regels is het iets anders: de berekende half4 van de regel staat in INK-kleur, en de onderste half4 staat in PAPER-kleur. Om de berekende regel te wijzigen, bereken je opnieuw de kleurcode, en vervang je de in deze instructie:

```
POKE 24771, (kleurcode)
```

Je kan best wat experimenteren met een aantal combinaties van kleuren, tot je er een hebt gevonden die voor jou het beste is. Het zal namelijk wel eens voorkomen, dat een bepaalde kleurcombinatie er op een ander manier toch anders uitkomt.

Heb je de beste combinatie gevonden, dan kan je die "vervangen" door het programma te laten lopen, en via het menu, Tasword op handinstructies te BASIC. Wanneer je de volgende naar het programma weer instelt, krijg je je aangepaste versie weer kleurrijk terug. (P.P.)



profiel software

onder andere toelaat om cartridges tot 500 K en meer te verspreiden.

Voor BL heeft Robert, in samenwerking met de huisprogrammeur van Micro-Connection, het "BL Business" pakket ontwikkeld. Het bestaat uit een Factoranalisprongende, klantentekst en voorraadbekend. De versie in Super Basic loopt goed. Sinds kort is er ook een geïntegreerde versie uitgebracht, die standaard sneller loopt. Omdat de zeer redelijke prijs 10000 fr., na P1400 heeft het pakket vrijwel alle mogelijkheden van professionals, en dus veel duurdere pakketten. Er is voorafweg geen Nederlandse versie beschikbaar, zodat het Nederlandse boekhoudsysteem toch wel verschilt van het Belgische.

Het BL Business-pakket is wel een echt Belgisch produkt: het programma spreekt twee talen, Frans en Nederlands, en je kan in de loop van het programma vrij kiezen en welke taal je aangesproken wordt te worden.

Over de BL zelf, weten beiden nog te vertellen, dat de voorraad ervan op de markt vrij kan afnemen. Er zouden nog naar een tienduizendtal verkopers exemplaren in de loop zijn, plus een half miljoen BL-"bonnen", in Korea. Een paar "broken" zijn druk bezig om de rechten van Astrad te kopen, waar het zit er dik in dat deze gekende zakenman het onderzet uit de kan wil. Afwachten wat het wordt, dus.

Micro-Connection werkt ook een paar maanden die het verspreiden wordt zijn.

Sinds kort is de "Music Machine" van RAN Electronics te verkrijgen. Die is te verkrijgen met de Specord van Cheeset. De Music Machine bevat een MIDI kaart, een piano, drum, een sampler en nog meer. Een beschrijving van deze muziek-machine volgt standaard binnenkort.



Van deze twee firma's hebben ze ook een Centronics interface uit. Ingevoerde tekstverwerker is de Soft-Print. Bij langzame niet zo uitgebreid als Terminal, maar alle echt noodzakelijke command's zitten er in. Ideaal voor iemand die af en toe 'n beetje op de printer wil hebben.

Micro-Connection werkt ook een Nederlandse vertaling van Masterfile (van Campbell Systems). Dat die de laatste tijd niet zo goed loopt, wijten ze - terecht vinden wij - aan het feit dat iedereen wel een versie van Masterfile heeft, sinds die gratis worden uitgeleverd bij het Interface 3-pakket.



Reest de relatief grotere systemen, van de BL Micro-Connection ook klein-waarschuwen vinden. Zij zijn inderdaad België de alleenverantwoordelijke voor de P1000 Engelse. Die lijkt wel een overgrote met-rekenmachine, maar blijkt bij nader inzien toch wel heel wat en zijn naar te hebben: 256 K RAM, en aansluitbaarheid op de BL, zijn behoorlijk naar huis. Wij kunnen roep onder de indruk van deze krachtige rekenmachine. Het systeemkond wij het vinden, kan je elders in dit nummer lezen.

Het planierige van deze klein opende ook die toch een grote uitgebreide assortiment heeft, is dat je er rustig binnen kunt stappen, zonder meteen lastig gevallen te worden door 'n verkoper. Er staan continu een aantal computers getrukket, waarvan de vele jeugdige bezoekers graag gebruik maken om game even het nieuwste spelletje uit te proberen. Een probleem voor Jacques of Robert. Dit vrij-bijzondere van de zaak, samen met de altijd vriendelijke en bereidwillige dienstverlening, vinden wij persoonlijk de twee grootste troeven van Micro-Connection.

Micro-Connection
BL - Katedalstraat 14-18
2000 Antwerpen
tele 03/231 01 03



DE WERKING VAN DE MICRODRIVE

De microdrive zijn nog steeds een veel gebruikt medium om programma's en data op te slaan. Meestal er de laatste tijd een aantal nieuw en betrouwbaardere opslag-systemen zijn bij gekomen. Discettes van 5 1/4 en de Beta-Disk INTERFACE, worden de microdrives nog door velen het plezier gebruikt.

Niet alle gebruikers van de INTERFACE 1 en de MICRODRIVE weten hoe deze werken. De hen nu enig inzicht te geven in de werking en opbouw van het systeem zal ik in een aantal afleveringen proberen het een en ander te verduidelijken.

Het zal een tijdsittige behandeling verhaald worden, maar daar ik nu vooral niet onderuit te lopen, zal ik de werking van deze schakels behandelen goed willen begrijpen.

1. De principes van INTERFACE 1

De allereerste vraag die het systeem opkomt is hoe het mogelijk is dat er een extra ROM van de ROM's toegevoegd aan het geheugen van de SPECTRUM 48K en 128. Zoals we weten kan een 680 processor "leest" 64 K adressen (0 tot 65535) en alleen 64 K adressen (65536 tot 131071) is geen redenerend kan er dus niet nog een extra ROM van de INTERFACE bij of wat meer?

In het gedeelte dat nu de ROM en 128 K ROM bevat kan niet gaan de ROM naar bij. Maar door een speciale truc, die we "paging" noemen kunnen we een parallel ROM inschakelen.

Dit paging wordt veel bij computers toegepast en het interne geheugen uit te breiden met extra banken RAM of ROM. De ROM's worden aangeproven als de ROM van de 2 K ROM schakelen hoewel de meeste processor hebben als drie Spectrums, de 680. Ook hier is paging toegepast en de diverse "slots" aan te spreken.

Bij paging hebben de ingang van de ROM's ook toegepast en INTERFACE 1 in te schakelen. "Paging" betekent letterlijk "pagina's". Men beschouwt nu de in de Spectrum ingebouwde ROM als een pagina en de ROM van INTERFACE 1 als de tweede pagina.

Bij paging wordt nu meestal de 128 K BASIC ROM en de de INTERFACE ROM aangeproven.

Veel computers gebruiken voor het "pagen" een aparte chip en de diverse pagina's aan te roepen, bij deze Spectrum geschikt

serie: microdrive

het "pagen" door een herleiding van de 128 K ROM af te vragen en daarna direct de de ROM van de INTERFACE 1 in te schakelen.

De werking van "paging" op de SPECTRUM

Op blz. 144 van het Engelse Handleboek van de Spectrum staat onderaan een tekening van de adresbus, waarop de INTERFACE aangesloten wordt. Op lijn 200onderhand printt het de ROM's van de Chip Select pin naar buiten.

Wanneer nu op deze lijn "0" staat, dan zal de interne 128 K ROM aangeschakeld worden. Het betekent nu er dan voor zorgt, dat er beschikbaar zijn een andere ROM wordt aangesloten, zodat andere de computer zal "lezen". De andere ROM kan een programma in ROM zijn (zoals in INTERFACE 2) of een andere BASIC-interpretatie zoals de de ROM van de INTERFACE 1.

De praktijk bij INTERFACE 1

Wanneer nu door de processor een commando gegeven, dat de 128 K BASIC-interpretatie niet kent, bijvoorbeeld "GOTO 1", dan wordt er beschikbaar een herleiding gegeven door een ROM, die deze concept. In nu geen INTERFACE 1 aangesloten, dan krijgt u deze herleiding op het scherm.

In er met een INTERFACE 1 aangesloten, dan controleert deze chip de adresbus van de Spectrum.

Wanneer nu een herleiding door op adres op dan schakelt de Interface 1 direct de interne ROM uit en schakelt zijn eigen de ROM in door eerst een "0" staat te zetten op ROM's en daarna beschikbaar zijn eigen de ROM aan te roepen.

De de ROM controleert dan of het ingevoerde commando er aan te die tijd kent. In een geval herkent hij "GOTO 1" en de herleiding wordt beschikbaar uittevoerd. Daarna zorgt de de ROM ervoor dat de 128 K BASIC-interpretatie weer ingeschakeld wordt zodat u verder kunt met de BASIC programma's.

Tot over "paging" met INTERFACE 1. Nu gaan we naar bekijken hoe de de ROM op programma's worden ingeschreven naar een MICRODRIVE CANTONING.

2. DE MICRODRIVE

HET MICRODRIVE DATA FORMaat

De te begrijpen hoe een microdrive werkt, moeten we eerst een paar definities op stellen wat de de gegeven ingeschreven.

Een microdrive is niet meer en niet minder dan een heel kleine harddisk, die een microdrive hard gebruikt. De gegevens worden op twee manieren tegelijk ingeschreven op de betrouwbaarheid te verbeteren.

microdrive

Voordat gegevens op een cartridge weggeschreven kunnen worden, moet de cartridge EERST GEFORMATEERD worden.

Dit FORMATEREN is nodig om de gehele band voortzetting in blokjes (SECTOR's) van 512 512 bytes langte te verdelen. Alleen blokjes band worden hierbij overgelegd, zodat afhankelijk van de beschikbaar van de band de opslagcapaciteit van een microdrivecartridge nogal sterk kan variëren.

Je zult zelf al wel gemerkt hebben dat bij intensief gebruik de cartridges nogal snel slijten en dat daardoor de toepassing van de microdrive in een stabielere sfeer er niet aantrekkelijker op.

De SECTOR's worden genummerd van 0 tot 255. Elke sector bevat ook nog een aantal identificatiegegevens, zoals een HEADER blok van 27 bytes.

De lege SECTOR bevat alleen dit HEADER blok en heeft dan nog een vrij DATA blok van 240 bytes over.

De 27 bytes van het HEADER blok zijn als volgt berekend:

12 bytes van ingangesignaal	
1 byte FLAG (0 of 1)	NOFLAG
1 byte SECTORnummer (0-255)	SECTOR
2 bytes niet gebruikt	
10 bytes CARTRIDGenummer	CARTRIDG
1 byte CHECKSUM (controlebyte) NOCHK	

Rest van de SECTOR vrij voor DATA blok

Schrijven nu een programma weg naar microdrive met het "SAVE P" ("P"=file name) commando dan zullen er afhankelijk van de lengte van het programma een aantal SECTOR's vol geschreven worden.

De SECTOR's bevatten nu een DATA blok, dat een RECORDHEADER en een RECORD bevat. Een DATA-blok bevat in totaal 240 bytes, die als volgt zijn berekend:

De RECORDHEADER:

12 bytes van ingangesignaal	
1 byte FLAG (0 of 1)	RECFLAG
1 byte RECORDnummer (0-255)	RECORDN
2 bytes RECORD lengte	RECLEN
10 bytes FILEnaam	FILENAME
1 byte CHECKSUM (controlebyte) NOCHK	

DATA:	
512 bytes DATAFILE (0-255)	RECORD
1 byte CHECKSUM (controlebyte) NOCHK	

De niet meteen dat er een duidelijke overeenkomst is tussen de 27 bytes van het HEADER-blok en de eerste 27 bytes van de SECTOR. De opbouw is nl. exact gelijk, dit is door KLONAR 10 gedaan zodat dat dezelfde RECFORMaat beide HEADERS kan lezen. (Bijspeling aan RLP-gegevens en andere data over 1)

Nu we weten hoe het programma per SECTOR wordt opgeslagen gaan we ons kijken hoe een programma nu op cartridge komt te staan.

Stel we hebben een programma "testen" dat uit 2240 bytes bestaat. Dan moeten we op de cartridge in 11,20 geheel op te slaan 2240/200=4,4 SECTOR's volaanvullen. Maar omdat we geen halve SECTOR's kunnen gebruiken moeten we dan 5 hele SECTOR's gebruiken, waarvan de laatste dan slechts gedeeltelijk "vol" is.

Dus: FILE "testen" zal op cartridge bestaan uit 5 RECORD's.
De RECORD's worden genummerd 0-4.

LET WEL: Dit RECORDnummer heeft NIETS uit te staan met het SECTORnummer.

De kan bijv. de FILE "testen" als volgt op de cartridge staan te staan:

RECORD	RECORD	RECORD	RECORD	RECORD
0	1	2	3	4

SECTOR	SECTOR	SECTOR	SECTOR	SECTOR
10	12	14	16	20

Je ziet dat de RECORD's NIET in opvolgende SECTOR's worden opgeslagen. Dit komt omdat de afstand van de RECORDIVE groter is dan de afstand waarmee de DATA van SECTOR tot SECTOR meer wordt volgeschreven met de volgende 512 bytes van de FILE. Door deze afstand is de band de volgende vrije SECTOR alweer voorbij als de computer naar de cartridge wil gaan schrijven.

Als de computer kan weten of een SECTOR vrij is, zal in de volgende aflevering uitlezen. Nu moet nog enkele opmerkingen over hetgeen in deze aflevering werd besproken.

We hebben gezien dat bij het formatteren er 256 SECTOR's gemaakt kunnen worden. Dit hebben we gezien, dat er per SECTOR een RECORD met een lengte van 512 bytes kan worden opgeslagen.

Dit zou dus betekenen dat er per cartridge maximaal 256*512=131072 bytes kan worden opgeslagen. Dit is 1280bytes, maar omdat de band die gebruikt wordt op cartridges van te maken niet voor 100% geschikt is de gegevens op te slaan, komt men in de regel niet veel verder dan 1100. Hoewel er geadviseerd werd met 100% opslagcapaciteit per cartridge, blijkt dat in de praktijk de meeste recente cartridges die 100% niet halen. Om bij voldoende gebruik zal door slijtage van de band de totale opslagcapaciteit langzaam teruglopen. Maar daarom ALTIJD BACK-UP COPIES VAN JE PROGRAMMA'S ***** E.A.



serie: peek en poke

De ZX Spectrum Systeemvariabelen Deel 2

23400 tot 23423

De eerste twee bytes van deze groep bevatten het registernummer waarnaar een GOTO of GOSUB-statement verwijst. De derde byte bevat het nummer van het statement binnen die regel. De vierde en vijfde byte bevatten het nummer van de regel die automatisch wordt uitgevoerd, en de zesde byte geeft het nummer van het statement op die regel dat nu wordt uitgevoerd.

Om dat het commando

GOTO regel 3, statement 7"

te interpreteren, doe je, direct na elkaar, als 1 commando, gevolgd door "ENTER", effect op een programmeerregel:

```
POKE 23400, 0-234000/256:
POKE 23401, INT (23400/256)
POKE 23402, 7
```

De na een GOSUB terug te keren na een andere regel dan de "roepende", bij een RETURN, kan je VOOR het GOSUB-commando, het registernummer waarnaar je wilt terugkeren bij het RETURN-commando, POKE in 23420 en 23421. De waarden van deze twee (en die van 23423, het statement-nummer) worden automatisch bijgehouden wanneer een GOSUB-commando wordt uitgevoerd, en te weten waarheen de RETURN voert.

23424 : BORDER

Deze byte bevat de attributen voor de border (0 voor het INPUT-gedeelte van het scherm, de daar volgende 4 bits 0, 1 en 2 bevatten de attributen bits 3, 4 en 5 de positieve en de border bits 6 de helderheid en bit 7 de toestand van flash. Wanneer je een bepaalde waarde wilt POKE, bereken je die door de een te nemen van: $100 + (0 \leq \text{PAPER}) + (64 \leq \text{BRIGHT}) + (128 \leq \text{FLASH})$.

Een POKE heeft niet onmiddellijk effect op de border: het kan eerst via een apart BORDER of CLR-commando. Een INPUT-statement zal wel direct de attributen voor de concrete twee regels overnemen, en zodra op ENTER gedrukt wordt, zal ook de BORDER de geforceerde kleur aannemen.

23425 en 23426 : PROG

Op deze twee adressen wordt het adres opgeslagen van het registernummer van de eerste regel van het Basic-programma, met andere woorden: op deze plaats vind je waar het programma-geheugen begint.

Door op het adres dat je in deze twee systeemvariabelen vindt, en het volgende adres, een 0 te POKE, geef je aan die eerste regel het nummer 0. Dit is niet op de gewone manier naar behouden te halen. Probeer maar.

Bij een standaard Spectrum, zonder Interface 1, is dit adres altijd 23760. Het Interface 1 aangesloten, verandert het, afhankelijk van het aantal "channels" dat aan "channels" gebonden wordt, en van bewerkingen met de micro-drive (die samen meestal ook geheugenruimte in beslag). Dus best eerst PEKEN voordat je POKE...

23427 en 23428 : INTEND

Hiervan slaat het adres van het karakter dat volgt op het laatste getal van de datum: een ENTER-code (13) of een andere punt (afschending van het volgende statement op die regel). Dit kan, met enig nadenken, gebruikt worden om een PASTORE uit te voeren naar het einde van een BEEK-regel. Het lijkt en anderszins toch wel gemakkelijk om de DATA waarnaar "pastored" wordt, op een aparte regel te zetten en geven het commando "PASTORE on" te gebruiken. Maar meestal gaat dat...

23441 en 23442 : K_LIN

Door deze adressen te PEKEN, weten we hoeveel geheugen het totale programma (tweede gebruikt variabelen inclusief, als we van de hier genoemde waarde, de waarde afrekken die we vinden in PROG. Vanaf het adres dat in deze twee systeemvariabelen staat, slaat de Spectrum de directe commando's op, of de programmeerregel die wordt ingetypt.

23443 : FLAG

Het enige nuttige gebruik voor de programmeur, is te vervullen een programma of via een commando, de hoofdletters aan te zetten.



profiel hardware

De Spectrus Doctor is een defecte handig instrument, dat kleinhandels en gebruikers een hoge tijd - en dus geld - kan besparen, door het goed ogenschijnlijk kan onderzoeken waar precies de fout zit, bij een defecte Spectrus. Het maakt het mogelijk te stellen een aanschaf te geven van de prijs van een eventuele reparatie. Daardoor kan de gebruiker ook beslissen of hij het de meeste waard vindt om de reparatie uit te laten voeren. Het apparaat toont ook aan of de Spectrus inderdaad wel defect is. "Geeft 'detector' kunnen duidelijk wel zien van de programmeerkunst van de gebruiker, of aan een defect cassettebandje liggen ...

Het apparaat - dat de "and voor", u en ik dus, 8000 Fr. (ca. 10000) kost - is niet alleen bedoeld voor de Spectrus-gebruiker zelf. Maar voor elke Sinclair dealer is het zo goed als een must. In Engeland hebben ze dat al een tijdje door 1 jaar werden al 200 van de apparaten verkocht.

Het randapparatuur voor de QL verlangt, heeft The English Computer Shop zich noodgedwongen beperkt. Omdat er voor die computer zoveel verschillende versies van de ROM in gebruik zijn. Als die interface voor de QL. Ik heb in Genua aan. Niet de meest geschikt is, maar hij is mogelijk de meer betrouwbare. Het de voorraad behoren ook de software van Microsoft, die kan inmiddels al jaren bestaan, en Philips doet natuurlijk ook mee. Janytisch-adaptors en een Commodore interface mogen eveneens ontbreken. Alleen de software voor de QL blijft problemen op te leveren, in die zin dat er altijd te weinig in voorraad is 1 alle software die uit Engeland wordt geïmporteerd, is binnen de kortste keren uitverkocht.

Deels de die computer tijd hoge leeftijd van de Spectrus. Blijven de gebruikers ervan nog steeds vragen aan randapparatuur, voornamelijk Centronics interface en joystick interfaces. Ook een vervangend toetsenbord blijft op het verlanglijstje. Het staan van vele rubber-toetsen, vooral daar de toetsen, door slijtage zijn gesloten, beperkt is. Erkel het goedkope aan degeijde toetsenbord van de Tronix wordt nog in voorraad gehouden. Volgens de heer Mihajlovic bleek uit de praktijk dat weinig mensen nu hebben de een goedkope computer met een toetsenbord uit te ruilen, dat bijna evenveel kost als de machine zelf. Een tijd voor de fabrikanten van toetsenborden ...



De nieuwste spruit van Sinclair/Attract, de Spectrus +2 wordt natuurlijk ook in voorraad genomen, van 10000 en vraag naar te in van 10000. Hij beschikbaar is, 10000000 dat gesproken had plaats werd verbeterd.

Een van de redenen waarom de Sinclair game shoot zo is doorgegaan als het getend, is volgens de heer Mihajlovic de invoerpolitiek. Bijna constant waren daar problemen met 1 te duur, geen voorraad, dus parallel-import. De problemen zijn volgens de heer Mihajlovic terug te voeren op de invoerder. Een waarheid als een koe, vinden wij 1 als de verdeler zijn job goed doet, is niemand geïnteresseerd in parallel-import.

De heer Mihajlovic stelt, om inzien wilken terecht, dat de handelaars die ingekouden zijn met Sinclair verkopen, goed moet zitten. Er is nog steeds een grote markt voor Sinclair producten, zeker wat software en randapparatuur betreft. Het is niet omdat de computer zelf niet meer zo veel wordt verkocht, dat de gebruikers ervan plots een uitgaatsvraag soort zijn geworden. Als voorbeeld heide hij aan, dat er nog steeds vraag is naar software voor de 8000 of de 10000. Dat zijn toch niet bepaald computers die in reusachtige hoeveelheden circuleren.

Verstandige handelaars hebben nu nog de kans de tot invoer te komen. Ook de grote, die toegesigd moeite hadden de Sinclair producten te vinden, hebben nu de eerste twee nummers van dit blad, en het verspreiden van de Spectrus +2 op de Nederlandse en Belgische markt, goed weten waar. Buiten de Nederlandse rond de computer 171 met de 1, zijn er ook een groot aantal mensen die graag nog randapparatuur en de nieuwste software voor hun Sinclair willen blijven kopen.

SCIENTIFIC/The English Computershop
Molen Spierplaats 2
2000 AMSTERDAM
Tel: 020-2330700

SPECTRUM
LOGICA

De Spectrum kent drie logische functies: AND, OR en NOT. Alle drie kunnen we op getallen worden toegepast. Als een AND kan ook met strings werken. Ook de gewone vergelijkingsovereenkomsten kunnen als logische beweringen gebruikt worden, zoals met getallen als met strings, met uitzondering van de beperking dat bij stringvergelijkingen, het resultaat een getal is.

De Spectrum herkent bij het werken met uitdrukkingen, een aantal regels. We hebben gebaseerd op een overzicht van de diverse mogelijkheden in een tabel samen te vatten. In deze tabel staat N een van nul verschillend getal voor.

1. De AND-functie

De logica die de Spectrum toepast in BASIC wordt af van de logica in machinetaal. Als je in BASIC bijvoorbeeld vraagt : PRINT 128 AND 1, dan krijg je 128 als resultaat. Een je dezelfde bewering in code : LD A,128 /AND 1, dan krijg je 0. Dit komt, doordat de AND-functie in code stelt dat per bit bekijken. Als een bit in byte 1 staat is 1, het de het overeenkomstige bit in byte 2 ook, dan wordt het overeenkomstig bit in het resultaat-byte overgenomen. In alle andere gevallen wordt het bit in het resultaat-byte gezet tot 0 gezet. Een illustratie van de voorgaande bewering :

```
128 = 1 0 0 0 0 0 0 0
1   = 0 0 0 0 0 0 0 1
AND = 0 0 0 0 0 0 0 0 = 0
```

Een ander voorbeeld : 129 AND 10

```
129 = 1 0 0 0 1 0 1 1
10  = 0 0 0 0 1 0 1 0
AND = 0 0 0 0 1 0 1 0 = 10
```

De laatste rijvert in BASIC het antwoord 129 op, in machinetaal is 10.

Als je in BASIC echt ANDen, dan kan dat via een for-1 machinetaal-variante :

```
LD A, X      43, X
AND Y      230, Y
LD B, 0      4, 0
LD C, A      79
RET        204
```

Op de plaats van X en Y, FORC je de 11 byte waarden die je wilt ANDen, en je voert uit met het commando : LET C = USR NN, waarbij NN het startadres is van de korte routine. Als kan je op het even naar in het geheugen laden. Je is zeker

with bytes lang, en past dus precies in een 100-letter, bijvoorbeeld de letter "A". Dan kan je in een programma de regel schrijven : LET C = USR USR "A". Vooral dan je : FOR USR "A"=1, 1 : FOR USR "A"=3, 7.

In BASIC kijkt de Spectrum enkel naar het "aantal"getallen van de tweede term in de functie : AND "Y". De tweede term is de functie : AND "Y". De tweede term wordt als logisch "waar" beschouwd, indien hij verschilt van nul. Hij wordt als logisch "falsch" beschouwd, indien hij nul is. In een term "waar", dan krijgt hij de logische waarde "1". In een term "falsch" dan krijgt hij de logische waarde "0". Verreken, in de functie : AND Y, de tweede term van nul, dan wordt de waarde van de functie gelijk aan de waarde van X, in het andere geval wordt de waarde van de functie nul.

De tweede term van de AND-functie kan ook bestaan uit een uitdrukking. De kan de Spectrum gemakkelijk een programma-regel versieren zoals :

```
LET C = A AND (B=12) AND (D=14)
```

of deze :

```
LET C = A AND (B="HALLO") AND (C < 20)
```

Een dergelijke regel wordt in twee tijden verwerkt. Eerst wordt de uitdrukking tussen haakjes verwerkt. De krijgt een waarde 1 of nul. Daarna worden de twee termen met elkaar ge-AND, en het resultaat daarvan wordt aan C toegewezen.

Een systeem kan je uitbreiden met gewone AND-functies als je wilt. Indien je een paar van twee AND-functies hebt, zal het eind-resultaat enkel 1 zijn, indien aan alle voorwaarden tegelijkertijd wordt voldaan. Als ook maar 1 van de 20 voorwaarden niet vervuld is, zal het resultaat van de reeks, nul zijn.

Een AND-functie kan je over het algemeen vervangen door een IF-betekent. In plaats van : LET C = A AND B kan je ook schrijven : LET C = A : IF B = 0 THEN LET C = 0. Langer, en onschikkeriger indien je meerdere voorwaarden hebt.

Niets heeft alleen maar voordelen : de IF-constructie heeft als nadeel, dat je verder op die regel rechte meer heeft kan en dat de eind-variabele C twee keer wordt gedefinieerd. Dat kost tijd. Je heeft dan meer het voordeel dat, indien je een regel kunt construeren in deze vorm : IF A = 10 THEN IF B = 10 THEN IF C = 7 THEN ... alle voorwaarden apart wordt bekken, en indien bv. de eerste voorwaarde niet vervuld wordt, het



programma meteen doorloopt op de volgende regel, terwijl in de AND-constructie, vooral bij de AND-combinatie apart wordt uitgetoetst voorwaar het andresultaat bekend kan worden.

De Spectrum staat ook een verkorte vorm van het IF-statement toe. In plaats van `IF A = 10 THEN ...` kan je ook schrijven `IF A THEN ...` zonder meer. Deze verkorte vorm kan je evenwel enkel gebruiken, indien je wilt testen of A al dan niet 0 is. Dus niet wanneer je wilt testen of A een bepaalde waarde heeft. De korte vorm van het IF-statement betekent dus eigenlijk `IF A <> 0 THEN ...`

De AND-functie werkt ook met strings. De inhoud van een variabele-string kan afleesbaar gemaakt worden van een bepaald tekenwaarde(n), door de formulering:

```
LET X = (INGET + AND Y(n)) * 255
```

Indien Y nuet een is, levert de eerste bare van de tabeling een lege string op (zie tabel); in het andere geval wordt de eerste bare "NIET".

Strings kunnen nu het even het bevatten, dus ook controletekens. Stel, je wilt een bepaalde tekst (T) in een bepaalde ink-tieler laten afdrucken op het scherm, bijvoorbeeld een kleur die bij een bepaalde speler hoort. Een bepaalde kleur het je in de tabel het statement:

```
LET X = ORS 16 + (ORS 2 AND SP = 1) +  
ORS 6 AND SP = 2)
```

of anderszels, met variabelingen:

```
LET X = ORS 16 + ORS 2 + 24SP = 2)
```

De tekst wordt dan voor speler 1 in het rood en voor speler 2 in het groen afgedrukt door het commando:

```
PRINT X;T;
```

Die reeks kan uitgebreid worden tot 8 kleuren (0 tot 7) voor 8 spelers, maar dan het het natuurlijk ook gemakkelijk door het commando `LET X = ORS 16 + ORS 8SP - 1`. Daarvoor krijgt elke speler de kleur die het bijhorende rangnummer overeenkomst. Dit doet niet, indien elke speler er niet op staat de een diegen kleur te kiezen. Dan moet je enkel de AND-reeks gebruiken, enkel acht regels met `IF SP = n THEN LET X = ...`

TABEL Logische functies

FUNCTIE	I	Y	RESULTAAT
X AND Y	0	0	0
	0	1	0
	1	0	0
	1	1	1
X OR Y	0	0	0
	0	1	1
	1	0	1
	1	1	1
NIET X	0	-	1
	1	-	0
X = Y	0	0	1
	0	1	0
	1	0	0
	1	1	1
X < Y	0	0	0
	0	1	0
	1	0	1
	1	1	0
X > Y	0	0	0
	0	1	1
	1	0	0
	1	1	0
X AND Y	0	0	0
	0	1	0
	1	0	0
	1	1	1
X OR Y	0	0	0
	0	1	1
	1	0	1
	1	1	1
X < Y	0	0	0
	0	1	0
	1	0	1
	1	1	0
X > Y	0	0	0
	0	1	1
	1	0	0
	1	1	0

FOR THE
SPECTRUM



SPECTRUM 128
COMPANIONSINCLAIR &
THE BURNING ISSUE
TECHNOLOGY

Iedereen die een aanvulling wilt op het Engelstalige handboek van de Spectrum 128 of de +2, of die wilt een praktische gids bij het programmeren van die twee computers, dan is dit boek een grappig instrument.

Ian Sinclair is niet een zeer profetaal, zonder onnodig technisch te worden, zonder de soms irritante verwijzingen naar "dat leer je later nog wel"-technieken die je in andere boeken vindt (het handboek bv.), maar de auteur de beginnende programmeur stap voor stap wegwijs in alle facetten van de 128 of de +2.

Alle mogelijkheden worden behandeld, besproken en met een of meerdere voorbeeld-programma's duidelijk gemaakt. De taal is eenvoudig. Het bespreekt de niet zo erg vaak bij te zijn in het Engels, kunnen met dit boek een eind ver komen, staat ik. De stijl is luchtig - wij laten te veel "jargon-onder-misbaar", maar dat is een kwestie van opzien.

Een handig stukje, veld is de voorbeelden die gegeven worden om eenvoudige en algemene berekeningen te zetten in BASIC uitdrukkingen. Voorbeelden zijn dit zeer simpel, voor anderen onoverwinnelijk. Ian Sinclair toont met enkele voorbeelden aan dat het ook echt eenvoudig is.

Ook de grafische mogelijkheden worden uitgebreid behandeld. Uit dit hoofdstuk blijkt dat je geen 256 kleuren ter beschikking heeft te hebben op je computer, de of zoals dingen een te kunnen kleuren...

Vooral hoofdstuk 10 is interessant. Daarin wordt nogal uitgebreid ingegaan op de voor Sinclair computers kleine speciale mogelijkheden. En die stellen nogal wat voor! De auteur is er duidelijk erg enthousiast over. En terecht. Het het PLAY-concept aan je van je Spectrum 128 of +2 een kleine synthesizer. Dat lijkt niet van de eerste keer, natuurlijk, maar aan de hand van de gegeven voorbeelden word je een heel eind op de goede weg geholpen.

Een aanrader voor ieder die met 31/12 128 of +2 meer wil doen dan spelletjes spelen. (P.P.)

Titel : Spectrum 128 Companion
Auteur : Ian Sinclair
Uitgever : Microtek Publishers Ltd
ISBN 1-85185-080-3
Prijs : £11.95, \$20 fr., ca. \$29,50

Ian Adams en Richard Kennedy zijn twee door de wet gewerde auteurs. De eerste schreef het handboek bij de Eric Audo, plus een werkboek over dezelfde computer (als kant die computer nog 91). De tweede schreef "The Complete Citi 44". Maar zich laat verscheiden een zeer dan boeken, de presentatie van het ding is eenzijdig gedomd.

Beide heren hebben nu samen een nieuw schriftje gepubliceerd. Over de opkomst en de ondergang van het Sinclair-koninkrijk.

De heren tonen ons tot de vertaling van de achterflap van het boek, die goed de auteur van het werk weergeeft. Wij vinden het geen aanrader.

"Sir Clive Sinclair, de zelf-made millionaire, standaardvoorbeld van de ondernemer, wordt soms opgevat als de manna van het kleine computer-nijverheid. De kritische kant die de ondernemerschap van de Engelse industrie aanvoelt. Ja toch."

"Dit ophefmakende en controversiële boek beschrijft de Sinclair-mythe van zeer dichtbij. De hele zaak aan één grote aflevering, onderhouden door het continue bewaken van de persconferenties, en in leven gehouden door een goedgelovige pers. Het verhaal van "Uncle Clive" en zijn bedrijven leert, dat de een helemaal geen oplossing had voor de miltgrijpende Britse industrie. Inderdaad: de Sinclair story heeft mogelijk geseit waarom de prestatie zo noodlottig is geweest."

"De manier waarop Sir Clive de grondrijke markt bemeesterde, samen met zijn steun over technologie, roepten uit in slechts bevoordeelde bedrijven. Dit was het enige politieke succes van een paar producten, die die nu op de markt van de toekomstige manier van zaken doen, leverde een reeks producten op die konden van technische ontwikkelingen en voor weinig of geen service werd verleend. De Sinclair-mythe blijkt een visioen te zijn, behalve als posthouder. Dit zegt alles over de bloederende van de liberale politiek in Engeland." (P.P.)

Sinclair & The Burning Technology, The Description of a Myth.
Auteurs : Ian Adams, Richard Kennedy
Uitgever : Penguin, Prijs : £3,70
ISBN 0-14-026774-8

EEN ABONNEMENT IS VOORDEELIGER

**Ik neem nu een abonnement op de SINCLAIR GIDS
op ontwerp 1 nummers voor fl. 44,00 ***

Name _____
 Street _____ No. _____
 Postcode _____ Place _____
 Met name van hantsoer _____

¹¹ Als ik voor 31 december 1986 een akkoord met een, wordt ik aangemerkt als „EASIS-Akkoord“ en zal ik kunnen profiteren van de voordelen van een EASIS-Akkoord.

Hi heb het aflossingsgeld overgemaakt op Giro 5406074 inv. Rindler-Gels.
Barbara Krieger

Voor België: 11 44,00 per Internationale Postzwaar van de Stedelijke Gede,
Postbus 111, 5100 AC Breda-Nieuw

IN/OUT

GRATIS LEZBESERVICE
www.lesb.org

Don't U see all these U lets men in
hundreds?

Plaats dan een annonce in de rubriek
198 (CULT)

JK ZDFK:



IK BIED AAN:



¹⁴ *Allgemeines KSET-commerce advantages werden gegeben*
Advantages our galspawds software of er case nation werden
ausgewandt

TE HOOF KWANTUM (orig. 1980 en 2007
aangepast tot stereoscopie). Amsterdam
aan M. Jansen, Buitenvoort 16, 1017
CZ Amsterdam (Holand).

© 2002 B. Williams, et al. *Journal of
Species Software*.
Eden et. Ping. Title No. 10.00 and
10.00-10.00.

IC 3000 Interface Information svt.
 reglen var komplett F&B-programm incl.
 Hardware lrv. Spectra.
 Tel. 0194-180

101. 100% Anteil-Dat. von Religionszugehörigkeit, 47 bis 48
E.L. in Japan, Harbinstr. 12, 2002 89
Bonn.

Beginner IBM? Also, an unprogrammed's
Level up cartridge also cassette. (origi-
nally). Another 1.8-inch, 1.2-inch, 1.2-
inch, 1.2-inch.

HC (200): Low cost version (available
also with Spectrum, HC-version of Lo-
Profile).
Tel.: 07231-7044

© 2004 contact met gebruikers van 020
Macquary met Spectrum 120 en eroverheen
met de afsluiter

U.S. Army, Department of the Army 241, 247 and
New York.

De 1000 programma's voor besturing van de VMEbus FPG's zijn uit de Spectra-Database worden vergeeft.
Tel: 040-200761

Wie helpt u? met de software van de
Fraction Lighten?

Frankrijk
als
bestaan

sinclair *gids*

Postbus 111
5111 AC Baarle-Nassau

Frankrijk
als
bestaan

sinclair *gids*

Postbus 111
5111 AC Baarle-Nassau

DE 3000: Radiofoon en telefoonapparatuur.
R. Berth, Oudekerk 43, Gronau a/d Rijn
Tele: 0663-70382

DE 1200: Tijdschrift voor Sinclair en II
en informatie van gebruikers.
A. van Bommel, Overveide 78, 3470 WB
Roosendaal
Tele: 0494-021947

DE 6000: Interdisc 1 + 2 accu-driven +
10 cartridges. Pk. 399,00
J. van Leeuwen, Leidsevaart 7, Heerlen
Tele: 02533-12508

DE 6000: HRC heeft belangstelling voor
uwer CASIOORG + 100000 rekenen.
J. Vrijthoff, Herengr. 39, 4201 BR
Breda

DE 6000: HRC heeft belangstelling voor
uwer CASIOORG + 100000 rekenen.
J. Vrijthoff, Herengr. 39, 4201 BR
Breda

DE 6000: II Spectrum 48K incl. software,
2 instructieboeken, Res. handleiding, Ver-
deling en handleiding, N.B.P. van Dijk - Vlaar-
dingen 830-432445 84 18,00 uur.



in/out

De redactie van Sinclair en HRC
wil graag tot medewerking van clubbe-
druften komen met andere verenigingen in
Nederland en België. Reacties aan:
G. Kuchelmeier, HRC-gids 84, 1704 BL Nieuw-
veen. Tele: 0206-61753

DATA-SKIP UW SINCLAIR-GIDS

Software, games

Paperboy	£ 25,-
Bomb Jack	35,-
Golden Rider	45,-
TV-racer	45,-
Tarzan	35,-
Knight Rider	35,-
Ass	45,-
Don Dink	45,-
Booby Busting	35,-
Sawhorse Fox	35,-
Mindbloss	35,-
Lightyear 1942	35,-
Star Trek II	35,-
Fat Worm	35,-
Omni 8 Options	35,-
Mind Vice	35,-
PBJ-chess	45,-
Dr. Adventure G	55,-
Arcade Creator	55,-

Aanbieding 1



DISCOVERY

fl 445,- met gratis
Terra Express

Software, series

Tekstbord II serie	£ 55,-
Tekstbord 2000/2001	45,-
Tekstbord 3 serie	45,-
Tekstbord 4 serie	45,-
Miro-QBASIC	35,-
OmniCalc	55,-
Strip 64 64 kolom	55,-
Art Studio	55,-
Mindbloss	55,-
Kata Basic	55,-
Laser Games	55,-
Laser Piano	55,-
Laser Compiler	55,-

Aanbieding 2



Gratis Corona per-printer

fl 399,-

Aanbieding 3



ZX Spectrum Plus Twee

fl 649,-

Software, Opus-disk

Tekstbord II	75,-
Tekstbord 3 serie	55,-
V. Hatch	35,-
OmniCalc spreadsheet	55,-
Superdisk 80 kol	55,-
Business Pack	75,-
Business Printing	55,-

Software, 128 K.

Tekstbord 128	65,-
Mind Vice	45,-
Golden Rider	45,-
Sawhorse Fox	35,-
Int. Matchday	45,-
Berry McG. Boxing	45,-
Technician Ted	35,-

Software Sinclair Q.L.

Chess 32	55,-
Blackpoint	55,-
Kerrie	55,-
Home Finance	55,-
Div. games	35,-
Brain Range	55,-
Academy	55,-
Mindful	55,-

Aanbieding 4



VTX-5000 modem + startpak

fl 149,-

Printers

Sinclair QL 11000 - 100 per	£ 650,-
Citizen LBP-10 per	750,-
Sony SPP-1000 per	750,-
Sinclair Corona F 80 per	350,-
Qprint Data Dinky per	450,-
Canberra RUC per/ser	550,-

Monitors

Green panel	250,-
Philips 7542, wit	350,-
Philips kleur	550,-

Diversen

ZX, printer interface	175,-
Sinclair Full-Screen Tr	250,-
Sinclair Word Terminal	
1000 Interface I	55,-
10 disks 2 1/2 inch-disk	45,-

Sinclair Q.L. hardware

Sinclair QL, complete	450,-
QST Disk Interface	345,-
Modem-pakket complete	345,-
RUC printer card	550,-
Monitors vari	250,-

Aanbieding 5



AMS-modem met software

fl 249,-

ZX-Spectrum Hardware

Multiple One	175,-
Multiple Designer	275,-
Saga Elite keyboard	250,-
Saga Two-Plus	250,-
10 profiles	150,-
10 2 1/2 inch diskettes	45,-
Translators cartridge box	75,-
Opus Discovery	450,-
met 10 gratis diskettes	
AMS mod. complete	249,-
VTX-5000 modem	149,-
Kerrie Speeding joyal	35,-

Dele-Skip is Sinclair-specialist en
levert binnen 24 uur alle de geboden
Sinxles

Buizen vóór het gebruik van de
QL zijn lange voor demonstratie

TEL 01820-20581

**Data-Skip,
L. Willemsteeg 10
2801 WC Gouda**

computercollectief

Amstel 312 (t.o. Canal) / 1017 JP / Amsterdam / Gpio 4 475 158 / Bank NMB 69 79 15 646

Wanneer u een van de volgende computers wilt kopen, wordt u verzocht de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

VERBODEN OM VERKRIJFT VAN NIEMAN EN ACTUELE BOEKEN VOOR DE SPECTRUM DE NL

BOEKEN, vers. de. SPECTRUM

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

BOEKEN, vers. de. SPECTRUM

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

ACTUELE EN NIET VERKRIJGBARE SOFTWARE

SOFTWARE, vers. de. SPECTRUM

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

SOFTWARE, vers. de. SPECTRUM

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

SOFTWARE, vers. de. SPECTRUM

De computercollectief is een vereniging van computerliefhebbers. Het is niet mogelijk om de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

Wanneer u een van de volgende computers wilt kopen, wordt u verzocht de afrekening te laten maken door de computercollectief. Het is niet mogelijk om de afrekening te laten maken door de computercollectief.

microcomputer geïmporteerd boeken en software